WinAgentLog Reference Manual

WinAgentLog License

Circitor (mailto:circitor@circitor.fr)

Table of Contents

Table of Figures

Chapter 1
Introduction

1.1 Preface

Welcome to the WinAgentLog reference guide!

This document describes how to configure and manage WinAgentLog. Background information for the technology and concepts used by the WinAgentLog product is also discussed.

1.2 What is WinAgentLog

The main goal of WinAgentLog is to collect and translate EventLog messages generated by the Windows EventLog subsystem and forward them to a host running a Syslog server.

WinAgentLog runs as a silent service with very little need for configuration to begin operating.

WinAgentLog service can operate on:

- Server: Windows 2000, Windows 2003 or Windows 2008.
- Workstation: Windows 2000, Windows XP, Windows Vista or Windows Seven.
- Hardware architecture: Intel x86 32 and 64 bits.

1.3 Main features

Main features of WinAgentLog service are:

- Can operate on any Windows version (Windows 2000, XP, 2003, Vista, Seven, 2008).
- Support Intel x86 32 and 64 bits hardware platform.
- Support both old "Windows EventLog API" and new "Windows Event Logging API" (introduced with Windows Vista).
- Windows EventLog messages are forwarded to the Syslog server in real time.
- WinAgentLog can remember the last Windows EventLog message sent and can restart from this point or from the end.
- Changes in the configuration file are automatically detected.
- Content of forwarded Syslog message is highly configurable.

1.4 Target audience and prerequisites

This document is intended for system administrators and consultants responsible for designing and maintaining logging solutions and log centers. It is also useful for IT decision makers looking for a tool to implement centralized logging in heterogeneous environments.

The following skills and knowledge are necessary for a successful WinAgentLog administrator:

- At least basic system administration knowledge.
- An understanding of networks, TCP/IP protocols, and general network terminology.
- Working knowledge of the Windows and UNIX operating system.
- In-depth knowledge of the logging process of various platforms and applications.

1.5 Organization of this document

This document is the reference manual of the WinAgentLog product. It explains:

- The basis of the Syslog protocol in the **Erreur ! Source du renvoi introuvable.**.
- The Microsoft EventLog subsystem in the Chapter 2.
- How to install WinAgentLog service in the Chapter 3.
- How to manage WinAgentLog service in the Chapter 4.
- How to configure WinAgentLog service in Chapter 5, Chapter 6 and Chapter 7.

1.6 Contacts and support information

Web site of the WinAgentLog product is:

- http://www.circitor.fr/winagentlog-en

Circitor team can be contacted by mail:

- mailto:circitor@circitor.fr

Web site of the Circitor is:

- http://www.circitor.fr

1.7 Version information

This guide is a work-in-progress document with new versions appearing periodically. The latest version of this document can be downloaded from the Circitor web site.

1.8 Feedback

Any feedback is greatly appreciated. General comments, errors found in text, and any suggestions about how to improve product and documentation are welcome.

1.9 Introduction to the Syslog protocol

The Syslog protocol is a network protocol widely used in the UNIX world. This protocol can carry log messages generated by applications on a host to another host. The purpose of this central host can be:

- Backup log messages for juridical aspects.
- Keep log messages in case of incident to perform investigation.
- Analyze log messages on the fly to perform real time correlation and detect incidents as soon as possible.

1.10 Syslog facilities

The Syslog protocol defines 24 numerical facilities. Some of the operating system daemons and processes have been assigned facility values. Processes and daemons that have not been explicitly assigned a facility may use any of the "local use" facilities. Those facilities that have been designated are shown in the following table along with their numerical code values.

Facility	Subsystem
0	kernel messages
1	user-level messages
2	mail system
3	system daemons
4	security/authorization messages (see note 1 below)
5	messages generated internally by syslogd
6	line printer subsystem
7	network news subsystem
8	UUCP subsystem
9	clock daemon (see note 2 below)
10	security/authorization messages (see note 1 below)
11	FTP daemon
12	NTP subsystem
13	log audit (see note 1 below)
14	log alert (see note 1 below)
15	clock daemon (see note 2 below)
16	local use 0 (local0)
17	local use 1 (local1)
18	local use 2 (local2)
19	local use 3 (local3)
20	local use 4 (local4)
21	local use 5 (local5)
22	local use 6 (local6)
23	local use 7 (local7)

Figure 1 Syslog facilities

- Note 1 - Various operating systems have been found to utilize Facilities 4, 10, 13 and 14 for security/authorization, audit, and alert messages which seem to be similar.
- Note 2 - Various operating systems have been found to utilize both Facilities 9 and 15 for clock (cron/at) messages.

1.11 Syslog severities

Each Syslog message also has a numerical severity level indicator. These are described in the following table along with their numerical values.

Severity	Signification
0	Emergency: system is unusable
1	Alert: action must be taken immediately
2	Critical: critical conditions
3	Error: error conditions
4	Warning: warning conditions
5	Notice: normal but significant condition
6	Informational: informational messages
7	Debug: debug-level messages

Figure 2 Syslog severities

1.12 Syslog messages over network

Syslog protocol is a text protocol carried over a UDP protocol. By default, the Syslog server is listening messages on the UDP port 514.

The Syslog protocol is completely defined in the RFC 3164.

Chapter 2 The EventLog subsystem

2.1 Introduction to the EventLog subsystem

The EventLog subsystem is specific to Microsoft Windows operating system. It is a central and standard method used by applications to log important events. EventLog is not really a protocol because it cannot be natively carried by the network.

2.2 The EventLog message

An EventLog message contains the following fields:

- The EventLog location.
- The EventLog type.
- The EventLog source.
- The EventLog category.
- The EventLog identifier.
- The EventLog message.
- Other information.

2.2.1 The EventLog location

By default, there are three locations where the EventLog messages are held. These locations are:

- Application: It contains EventLog messages logged by applications and services. For example, a database application might record a file error. The application developer decides which events to monitor.
- System: It contains EventLog messages logged by system components, such as the failure of a driver or other system component to load during startup. The events logged by the system components are predetermined.
- Security: It contains EventLog messages such as valid and invalid logon attempts, as well as events related to resource use such as creating, opening, or deleting files or other objects. An administrator can turn on auditing to record events in the security log.

Other location can be created and used if needed.

- For example, Internet Explorer 7.0 defines and uses the location named "Internet Explorer"
- Another example, Windows 2003 Server defines and uses the locations named "Directory Service", "DNS Server" and "File Replication Service".

The next image shows existing EventLog locations as displayed by the "Event Viewer" application in the left part of the window.

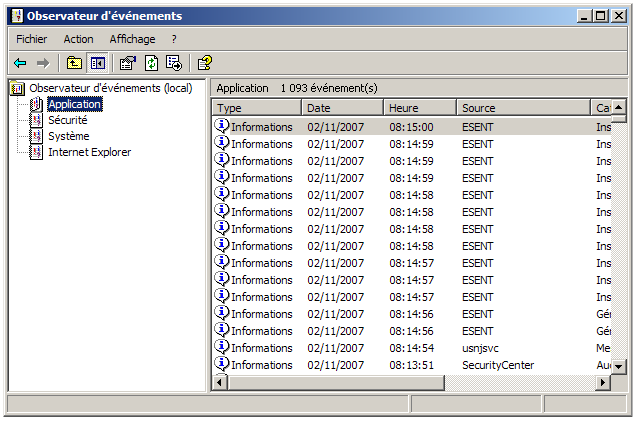

Figure 3 EventLog locations as shown by the "Event Viewer" application

2.2.2 The EventLog type

There are five EventLog types that exist. The "Event Viewer" application uses this type to determine which icon to display in the list view of the EventLog messages.

- Error: An EventLog message that indicates a significant problem such as loss of data or loss of functionality. For example, if a service fails to load during startup, an ‘Error’ event is logged.
- Warning: An EventLog message that is not necessarily significant, but may indicate a possible future problem. For example, when disk space is low, a ‘Warning’ event is logged. If an application can recover from an event without loss of functionality or data, it can generally classify the EventLog as a ‘Warning’ event.
- Information: An EventLog message that describes the successful operation of an application, driver, or service. For example, when a network driver loads successfully, it may be appropriate to log an ‘Information’ event. Note that it is generally inappropriate for a desktop application to generate an EventLog message each time it starts.
- Audit Success: An EventLog message that records an audited security access attempt that is successful. For example, a user's successful attempt to log on to the system is logged as an ‘Audit Success’ event.
- Audit Failure: An EventLog message that records an audited security access attempt that fails. For example, if a user tries to access a network drive and fails, the attempt is logged as an ‘Audit Failure’ event.

More information is available on the Microsoft MSDN web site:

- http://msdn2.microsoft.com/en-us/library/dhacse28.aspx

The next image shows EventLog messages with different EventLog types as displayed by the "Event Viewer" application in the right part of the window.

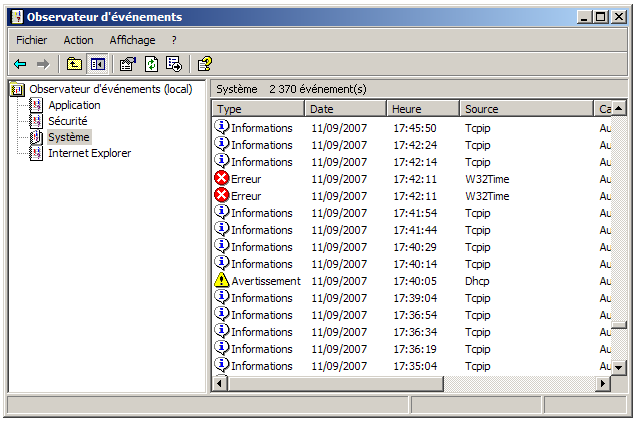

Figure 4 EventLog types as shown by the "Event Viewer" application

2.2.3 The EventLog identifier

The EventLog identifier is a 32 bit number that identifies the EventLog message. The "Event Viewer" application shows this EventLog identifier.

This number is laid out as followed:

- The "EventLog identifier code part" is a 16 bit number that identify a message.
- The "EventLog identifier facility part" is a 12 bit number. It can help to identify the subsystem or the context that generated the EventLog message. This facility is totally different to the Syslog facilities concept.
- The "EventLog identifier reserved bit" is a 1 bit number. It is reserved for a future usage.
- The "EventLog identifier customer bit" is a 1 bit number. It indicates if the EventLog message is generated by the system or by another application.
- The "EventLog identifier severity part" is a 2 bit number. It gives a severity level to the EventLog message. This severity level is totally different to the EventLog type concept.

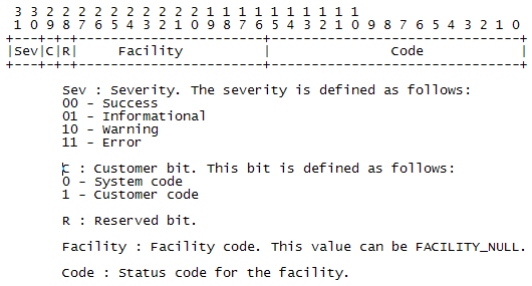

Figure 5 Parts of an EventLog identifier

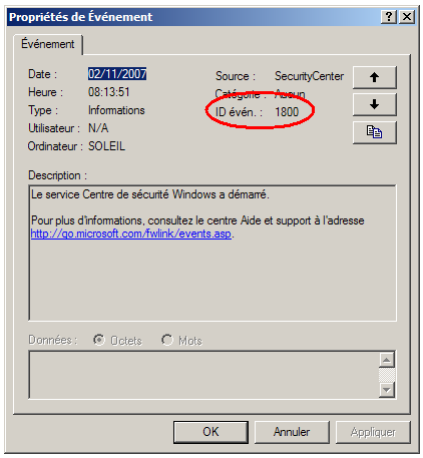

Figure 6 EventLog identifier as shown by the "Event Viewer" application

2.2.4 The EventLog source

The EventLog source is the name of the software that logs the event. It is often the name of the application or the name of a subcomponent of the application if the application is very large.

An application or a service can provide its own name. It can be "MSSQLSERVER" or "Tcpip" for example.

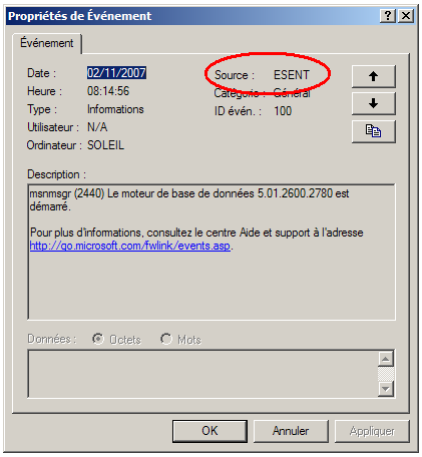

Figure 7 EventLog source as shown by the "Event Viewer" application

2.2.5 The EventLog category

The goal of EventLog category is to organize and add more information to EventLog messages. An EventLog category is a text string and each EventLog source can define categories.

The meaning of the EventLog category depends on the EventLog source. The "Event Viewer" application shows the category of an EventLog message.

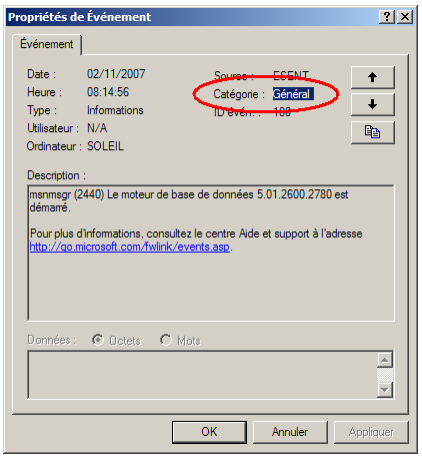

Figure 8 EventLog category as shown by the "Event Viewer" application

2.2.6 The EventLog message

The meaning of the EventLog message depends on the EventLog source. The "Event Viewer" application shows the message text of an EventLog message.

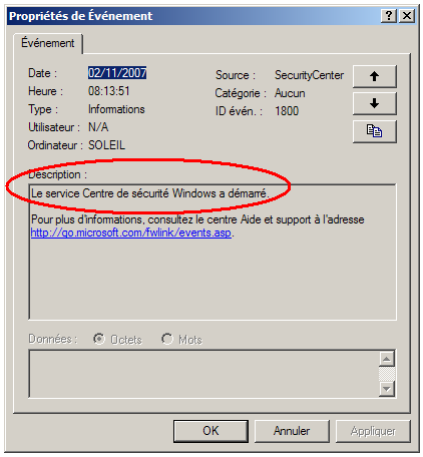

Figure 9 EventLog message as shown by the "Event Viewer" application

2.2.7 Other information

Less important information can be found in an EventLog message. It is also possible to get:

- The host name. It is the name of the host where the EventLog message has been generated.
- The user name. It is the name of the user that generated the EventLog message. This information may be not available if it has no meaning in the process context.
- The record number. This value is the number of the record in EventLog location database. It can be used to read or find for a specified record.
- Time generated. The time at which this entry was submitted. This value is a Universal Coordinated Time.
- Time written. The time at which this entry was received by the EventLog subsystem and written into the EventLog location database. This value is a Universal Coordinated Time.
- The record length. It is the size in bytes occupied by the EventLog message in EventLog location database. It is not the length of the message itself.

2.3 The EventLog subsystem and the registry

2.3.1 Root registry key

The root registry key used by EventLog subsystem is:

- HKLM\SYSTEM\CurrentControlSet\Services\EventLog

HKLM means the HKEY_LOCAL_MACHINE hive.

2.3.2 EventLog locations

All EventLog locations are listed in the registry under the "HKLM\SYSTEM\CurrentControlSet\Services\EventLog" registry key.

```
Example:
HKLM\SYSTEM\CurrentControlSet\Services\EventLog\Application
HKLM\SYSTEM\CurrentControlSet\Services\EventLog\Security
```

Each EventLog location registry folder contains also registry information specific the location, as shown in the following table:

Registry Value	Description
DisplayNameFile	This registry key identifies the file that stores the localized name of the EventLog location. The name stored in the specified file appears as the EventLog location in the "Event Viewer" application. If this entry does not appear in the registry for an EventLog location, then the "Event Viewer" application displays the name of the Log-name subkey as the EventLog location name.
DisplayNameID	This registry key stores the message identification number of the EventLog location name string. This number indicates the message in which the localized display name appears. The message is stored in the file specified by the value of the DisplayNameFile entry. For example, if the value of DisplayNameID is 0x5, then the localized display name for the EventLog location appears in Message ID 5. If this entry does not appear in the registry for an EventLog location, the "Event Viewer" application displays the name of the Log-name subkey as the EventLog location name.
File	This registry key specifies the fully qualified path to the file where each EventLog message is stored. This entry enables the "Event Viewer" application and other log processing programs to find the EventLog database files. The value of this entry is displayed in Log name in the Properties dialog box for a log in the "Event Viewer" application.
MaxSize	This registry key specifies the maximum size of the EventLog database files. This entry corresponds to the Maximum log size option on the Properties page for a log in the "Event Viewer" application.
PrimaryModule	This registry key stores the name of the subkey in Services\Eventlog\Log-name that contains default values for the entries in the Services\Eventlog\Log-name\Source-name subkey. These default values are used to display events in the "Event Viewer" application when the source program has not specified values in a Source-name subkey, or when files specified as the value of an entry cannot be opened or do not contain the required data. For example, in the Services\Eventlog\Security subkey, the default value of PrimaryModule is Security. This value indicates that the Services\Eventlog\Security\Security subkey contains default values for entries that appear in the Services\Eventlog\Security\SourceName subkey. These values are used to display events in the security log when the source program does not specify valid values for the entries.
Retention	This registry key specifies the time, in seconds, that records of events are protected from being overwritten. When the age of an event reaches or exceeds the value of this entry, it can be overwritten.

	• 0: EventLog subsystem overwrites EventLog messages only when the EventLog database file reaches the maximum size, as defined by the value of the MaxSize entry. This value corresponds to the Overwrite events as needed option on the Properties dialog box for a log in the "Event Viewer" application. • 0x1–0xFFFFFFFE: Specifies how EventLog messages are protected. • 0xFFFFFFFF: EventLog messages are never overwritten. When the EventLog database file reaches its maximum size, as defined by the MaxSize entry, you must clear the file manually by clicking "Clear all Events". Otherwise, the file discards new events. The EventLog database file must be cleared before any change to its size can take effect. This value corresponds to the "Do not overwrite events" option in the "Event Viewer" application. You can also activate a security feature that prevents a machine from being used if it cannot generate audit EventLog messages. To activate this feature, set the value of the CrashOnAuditFail entry to a value other than 0.
Sources	This registry key specifies the applications, services, or groups of applications that write EventLog messages to this EventLog location. Each program listed in the value of this entry is represented by a subkey in the Log-Name subkey.
AutoBackupLogFiles	Using this registry key causes the EventLog subsystem to automatically back up the EventLog database files and clear it. On computers with the "CrashOnAuditFail" policy turned on, the computer continues to log events (instead of hanging because of an audit failure) if the current EventLog database files can be backed up automatically.
RestrictGuestAccess	As EventLog messages can contain sensitive data this registry key allows you to restrict access to administrators and system accounts only. • 0: guest access is allowed • 1: administrators and system accounts access only

2.3.3 EventLog sources

All EventLog sources for a particular EventLog location are listed in the registry under the "HKLM\SYSTEM\CurrentControlSet\Services\EventLog\<location>" registry key.

```
Example
HKLM\SYSTEM\CurrentControlSet\Services\EventLog\Application\.NET Runtime
HKLM\SYSTEM\CurrentControlSet\Services\EventLog\SC Manager
HKLM\SYSTEM\CurrentControlSet\Services\EventLog\System\acpi
…
```

Each EventLog source registry folder contains registry information specific to the software that will be logging the EventLog messages, such as a message file, as shown in the following table:

Registry Value	Description
CategoryCount	Number of EventLog categories supported. This value is of type REG_DWORD.
CategoryMessageFile	Path for the category message file. A category message file contains language-dependent strings that describe the EventLog categories.
EventMessageFile	Path for the EventLog message file. You can list multiple files, separated by semicolons. An EventLog message file contains language-dependent strings that describe the EventLog message.
ParameterMessageFile	Path for the EventLog parameter message file. A EventLog parameter message file contains language-independent strings that are to be inserted into the EventLog description strings.
TypesSupported	Bitmask of supported types. It can be one or more of the following values: • EVENTLOG_AUDIT_FAILURE • EVENTLOG_AUDIT_SUCCESS • EVENTLOG_ERROR_TYPE • EVENTLOG_INFORMATION_TYPE

	• EVENTLOG_WARNING_TYPE

Chapter 3
Installing WinAgentLog

3.1 What do you need to install WinAgentLog?

The following OS and platform are supported to install and run WinAgentLog :

- Microsoft Windows 2000
- Microsoft Windows XP
- Microsoft Windows Vista
- Microsoft Windows Seven
- Microsoft Windows 2003 server
- Microsoft Windows 2008 server

Both Intel x86 32 bits or 64 bits hardware platforms are supported.

3.2 Installing WinAgentLog product

You must be logged in with an administrative account to install or uninstall WinAgentLog. To install WinAgentLog, simply run the "WinAgentLog_Setup.exe" program.

On Windows Vista, Seven or 2008 server, you may have to allow UAC to give privileges to the setup program.

3.2.1 Select language

First of all, you have to select the language to use during the setup procedure. English and French languages are supported during WinAgentLog installation:

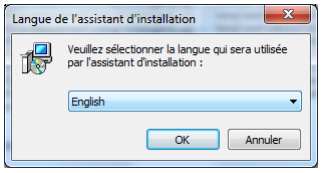

Figure 10 Setup language

3.2.2 Welcome screen

The first screen displays the version of WinAgentLog that will be installed by the setup:

Figure 11 Welcome screen

3.2.3 License

This screen displays the license of the WinAgentLog product. You have to accept the agreement before continuing:

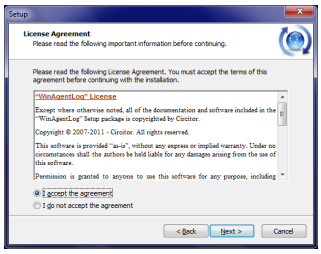

Figure 12 License

3.2.4 Destination folder

This screen lets you choose the destination folder where to install the WinAgentLog product. The default folder is "C:\Program Files\Circitor\WinAgentLog":

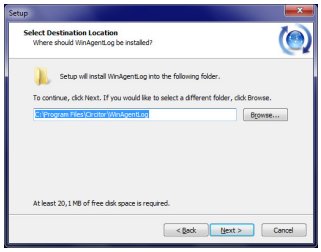

Figure 13 Destination folder

3.2.5 Syslog server

This screen lets you enter the IP address or the name of a host running a Syslog server:

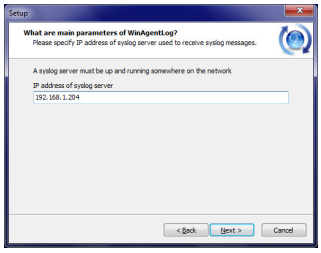

Figure 14 Syslog server

3.2.6 Start menu folder

This screen lets you choose the start menu folder where to install various shortcuts of the WinAgentLog product. The default folder is "Circitor\WinAgentLog":

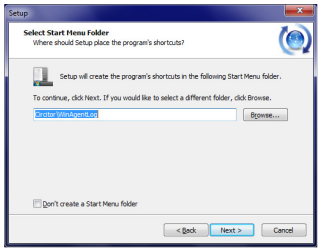

Figure 15 Start menu folder

3.2.7 Additional tasks

This screen lets you select to perform or not some additional tasks:

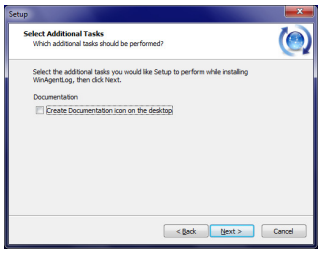

Figure 16 Additional tasks

3.2.8 Ready to install

This screen shows you a summary of tasks that will be performed by the setup program:

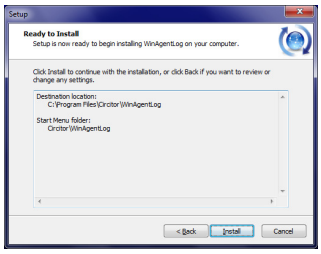

Figure 17 Ready to install

3.2.9 History version

This screen shows you the version history of the product:

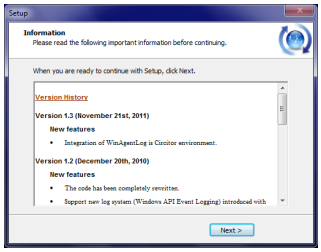

Figure 18 History version

3.2.10 Finish screen

This screen is displayed when the setup program has finished installing the product:

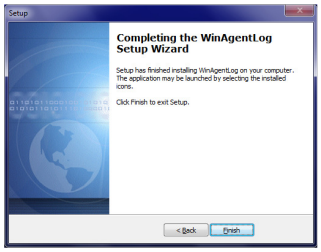

Figure 19 Finish screen

3.3 Installed files

3.3.1 Files installed on the disk

The root installation folder is by default "C:\Program Files\Circitor\WinAgentLog". It is possible to modify this path during the setup procedure.

At the end of the installation, the following files are installed on the hard disk:

- WinAgentLog.exe. It is the executable of the WinAgentLog service.
- Restart.exe. It is the executable needed by the WinAgentLog service.
- WinLog.dll. It is DLL needed by the WinAgentLog service.
- WinAgentLog.conf.xml. This file is the configuration file of the WinAgentLog service.
- WinAgentLog.conf.template.xml. This file is a template of the configuration file of the WinAgentLog service.
- unins000.exe. This file is the program to uninstall the WinAgentLog service.
- unins000.dat. This file contains extra data used by the uninstall program of the WinAgentLog service.
- Reference Manual.pdf. This document.

3.3.2 Modifications in the startup menu

At the end of the installation, a new entry is created in the startup menu. It is possible to disable the creation of this menu during the setup procedure. The following entries are created in the menu:

- Configuration file. It is a shortcut to the configuration file of the WinAgentLog service.
- Reference Manual. It is a shortcut to this document.
- Start Service. It is a shortcut to start the WinAgentLog service. On Windows Vista, Windows Seven and Windows 2008 server, it has to be run as administrator by right clicking on the menu item.
- Stop Service. It is a shortcut to stop the WinAgentLog service. On Windows Vista, Windows Seven and Windows 2008 server, it has to be run as administrator by right clicking on the menu item.
- Uninstall. It is a shortcut to completely uninstall the WinAgentLog service.
- WinAgentLog WWW site. It is a shortcut to a web page related to the latest news of the product.
- Circitor WWW site. It is a shortcut to company web site.

3.4 Extra installation

Depending on your configuration, the setup may want to install “Microsoft Visual C++ 2010 SP1 Redistributable”:

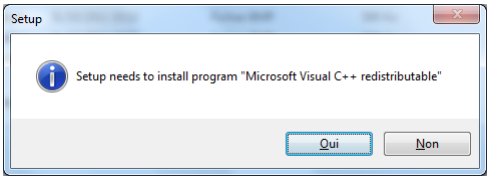

Figure 20 Microsoft Visual C++ 2010 SP1 Redistributable installation

You MUST accept installing this component.

Chapter 4
Managing the WinAgentLog service

4.1 Starting and stopping WinAgentLog service

There are three ways to start and stop the WinAgentLog service:

- By the Service Control Manager
- By the command line prompt
- By the start menu

4.1.1 By the Service Control Manager

Simply start the Service Control Manager. You can start and stop the WinAgentLog service in the same way as other services.

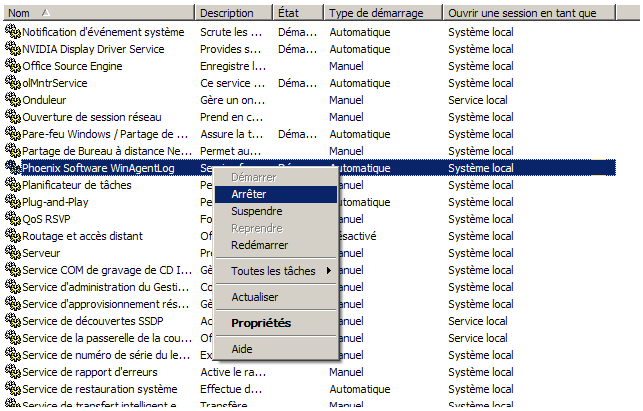

Figure 21 Manage the WinAgentLog service from the Service Control Manager

4.1.2 By the command line prompt

You can start and stop the WinAgentLog service by the command line:

- net start __WinAgentLog
- net stop __WinAgentLog

You can also start and stop the WinAgentLog service by the program itself:

- C:\Program Files\Circitor\WinAgentLog\WinAgentLog.exe --start
- C:\Program Files\Circitor\WinAgentLog\WinAgentLog.exe --stop

On Windows Vista, Seven and 2008 hosts, the command shell must be executed with administrator rights (right click on icon and choose "execute as administrator").

4.1.3 By the "start" menu

You can also start and stop the WinAgentLog service by the start menu

In the "start" menu, item "Program", sub item "WinAgentLog", there are 2 options:

- Start service
- Stop service

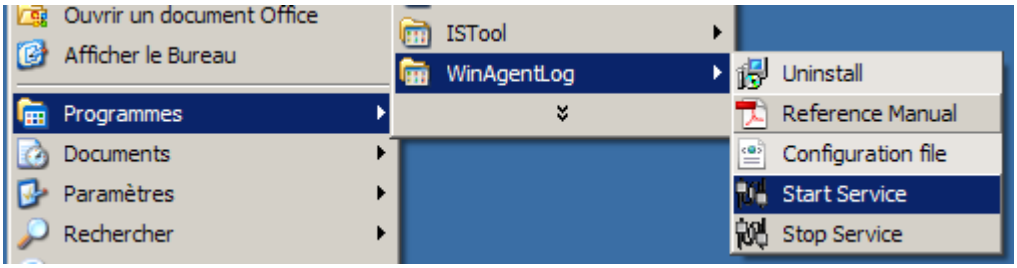

Figure 22 Manage the WinAgentLog service from the "start" menu

On Windows Vista, Seven and 2008 hosts, these commands must be executed with administrator rights (right click on icon and choose "execute as administrator").

4.2 Registry usage

All registry entries created and used by the WinAgentLog service are located under the following keys:

- HKEY_LOCAL_MACHINE\SOFTWARE\Circitor\WinAgentLog

The following keys are created or used by WinAgentLog service:

- Version: This key contains the version of the product.
- LastBookmark_<EventLog location>: This key contains information to remember the last EventLog message read for a specific EventLog location. It is used when the service is restarting to start to read new EventLog messages from this event. There is one key for each monitored location. If the key is not present for a given location, the service is starting from the end.

For now, the WinAgentLog service does not create nor use any other registry entries.

Chapter 5
Configuring WinAgentLog service

The configuration parameters of WinAgentLog service are stored in an XML file. The name of the configuration file is "WinAgentLog.conf.xml". This file can be found in the installation folder (C:\Program Files\Circitor\WinAgentLog by default).

When changes are made in the configuration file, they are automatically detected by WinAgentLog service which restarts immediately to apply changes.

If the configuration file has not a valid XML syntax, WinAgentLog service will be unable to parse it and will automatically stop. You can check the syntax validity of the configuration file by viewing it with an XML viewer (Internet Explorer for example).

5.1 Example of a configuration file

The following configuration file is the default one installed by the setup program:

```
<?xml version="1.0" encoding="UTF-16"?>
<Config>

   <EventLogList DefaultStartFromLastSend="Yes" DefaultDateFormat="%d/%m/%Y %H:%M:%S" DefaultFormat="$MESSAGE"
DefaultFacility="16" DefaultSeverityError="3" DefaultSeverityWarning="4" DefaultSeverityInformation="6"
DefaultSeverityAuditSuccess="7" DefaultSeverityAuditFailure="5">
      <EventLog Name="System" Enabled="Yes" Facility="16"/>
      <EventLog Name="Application" Enabled="Yes" Facility="17"/>
      <EventLog Name="Security" Enabled="Yes" Facility="18"/>
   </EventLogList>

   <SyslogServerList DefaultPort="514" DefaultProtocol="udp" DefaultMaxLength="1514" DefaultEndOfMessage="">
      <SyslogServer Name="127.0.0.1" Enabled="Yes"/>
   </SyslogServerList>

   <Heartbeat Enabled="Yes" TimeoutMs="60000" Facility="23" Severity="6"/>

</Config>
```

5.2 Parameters types

The types of the different parameters that can be found in the configuration file can be one of the following: string, integer, boolean.

- A String parameter can be empty or can have any arbitrary length of characters.
- An Integer parameter is a number. It can contain only digit characters and the sign plus (+) or minus (-).
- A Boolean parameter can have two values: "Yes" or "No".

5.3 The <EventLogList> parameter block

This block contains the parameters used to define the behavior of the WinAgentLog "EventLogList" feature.

5.3.1 Attribute "DefaultStartFromLastSend"

This attribute is a Boolean value that defines if the WinAgentLog service has to remember or not the last generated EventLog.

If set to "Yes", WinAgentLog service starts to monitor from the last generated EventLog. If set to "No", it starts to monitor from the end of the EventLog location.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "No" (begin from end of the location when WinAgentLog service starts).

5.3.2 Attribute "DefaultDateFormat"

This attribute is a string value that defines the format of date and hour to use when forwarding an EventLog messages. The code of the program uses the C function _tcsftime() (see http://msdn.microsoft.com/en-us/library/fe06s4ak(v=vs.80).aspx for more help). Any valid format as described in the MSDN page can be used.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "%d/%m/%Y %H:%M:%S" (French date and hour format).

5.3.3 Attribute "DefaultFormat"

This attribute is a String value that defines the format of a Syslog message generated by WinAgentLog service when forwarding an EventLog message. See more information about "message formatting" in the next chapter.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "$MESSAGE".

5.3.4 Attribute "DefaultFacility"

This attribute is an Integer value that defines the Syslog facility for Syslog messages generated by WinAgentLog service when forwarding an EventLog message.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "16" (Syslog facility code "local0").

5.3.5 Attribute "DefaultSeverityError"

This attribute is an Integer value that maps the EventLog "error" severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "3" (Syslog severity code "error").

5.3.6 Attribute "DefaultSeverityWarning"

This attribute is an Integer value that maps the EventLog "warning" severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "4" (Syslog severity code "warning").

5.3.7 Attribute "DefaultSeverityInformation"

This attribute is an Integer value that maps the EventLog "information" severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "6" (Syslog severity code "informational").

5.3.8 Attribute "DefaultSeverityAuditSuccess"

This attribute is an Integer value that maps the EventLog "audit success" severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "7" (Syslog severity code "debug").

5.3.9 Attribute "DefaultSeverityAuditFailure"

This attribute is an Integer value that maps the EventLog "audit failure" severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "5" (Syslog severity code "notice").

5.3.10 Folder "EventLog"

The WinAgentLog service is able to monitor multiple EventLog location at a time. You have to define multiple "<EventLogList/EventLog>" blocks, one for each EventLog location you want to monitor.

5.4 The <EventLogList>/<EventLog> parameter block

This block contains the parameters used to define the behavior of one of the WinAgentLog "EventLog" feature.

5.4.1 Attribute "Name"

This attribute is a String value that defines the name of the EventLog location to monitor. You can use the EventViewer program to list the EventLog location managed by your system.

Well known and historical locations are:

- Application
- System
- Security

If this attribute is not set, the current EventLog block is ignored.

5.4.2 Attribute "Enabled"

This attribute is a Boolean value that defines if WinAgentLog service is allowed to monitor the current EventLog location.

If this attribute is not set, the current EventLog block is ignored.

5.4.3 Attribute "StartFromLastSend"

This attribute is a Boolean value that defines if the WinAgentLog service has to remember or not the last generated EventLog for the current EventLog block.

If set to "Yes", WinAgentLog service starts to monitor from the last generated EventLog. If set to "No", it starts to monitor from the end for the current EventLog location.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultStartFromLastSend" is used.

5.4.4 Attribute "DateFormat"

This attribute is a string value that defines the format of date and hour to use in EventLog messages for the current EventLog location. The code of the program uses the C function _tcsftime() (see http://msdn.microsoft.com/en-us/library/fe06s4ak(v=vs.80).aspx for more help). Any valid format as described in the MSDN page can be used.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultDateFormat" is used.

5.4.5 Attribute "Format"

This attribute is a String value that defines the format of a Syslog message generated by WinAgentLog service when a new EventLog message is available for the current EventLog location. See more information about Message formatting in the next chapter.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultFormat" is used.

5.4.6 Attribute "Facility"

This attribute is an Integer value that defines the Syslog facility for Syslog messages generated by WinAgentLog service when a new EventLog message is available for the current EventLog location.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultFacility" is used.

5.4.7 Attribute "SeverityError"

This attribute is an Integer value that maps the EventLog error severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultSeverityError" is used.

5.4.8 Attribute "SeverityWarning"

This attribute is an Integer value that maps the EventLog warning severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultSeverityWarning" is used.

5.4.9 Attribute "SeverityInformation"

This attribute is an Integer value that maps the EventLog informational severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultSeverityInformation" is used.

5.4.10 Attribute "SeverityAuditSuccess"

This attribute is an Integer value that maps the EventLog audit success severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultSeverityAuditSuccess" is used.

5.4.11 Attribute "SeverityAuditFailure"

This attribute is an Integer value that maps the EventLog audit failure severity code to a Syslog severity code.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultSeverityAuditFailure" is used.

5.5 The <SyslogServerList> parameter block

This block contains the parameters used to define the behavior of the WinAgentLog "SyslogServerList" feature.

5.5.1 Attribute "DefaultPort"

This attribute is an Integer value that defines the destination port used by the remote Syslog server.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "514" (default port for Syslog protocol).

5.5.2 Attribute "DefaultProtocol"

This attribute is a String value that defines the layer protocol used by Syslog protocol. The only supported value for now is "udp".

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "udp" (the standard protocol used by Syslog).

5.5.3 Attribute "DefaultMaxLength"

This attribute is an Integer value that defines the maximum length of a Syslog message. Be careful with this value because the RFC 3164 (the Syslog RFC) clearly indicates that the maximum length of a Syslog frame should be lower than 1024.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "1024".

5.5.4 Attribute "DefaultEndOfMessage"

This attribute is a String value that defines special characters that can be added at the end of each Syslog message. The only supported values are:

- "": this empty sequence will add nothing at the end of the Syslog message.
- "CR": this will add a carriage return code (ASCII value \r or 0x0d) at the end of the Syslog message.

- "LF": this will add a line feed code (ASCII value \n or 0x0a) at the end of the Syslog message.
- "CRLF": this will add a carriage return code followed by a line feed code (ASCII values \r\n or 0x0d 0x0a) at the end of the Syslog message.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "" (empty string).

5.5.5 Folder "SyslogServer"

The WinAgentLog service is able to forward log messages to multiple Syslog servers at a time. You have to define multiple "<ServerList/Server>" blocks, one for each Syslog server where to forward log messages.

5.6 The <SyslogServerList>/<SyslogServer> parameter block

This block contains the parameters used to define the behavior of one of the WinAgentLog "SyslogServer" feature.

5.6.1 Attribute "Name"

This attribute is a String value that defines the name or the IP address of a remote Syslog server.

If this attribute is not set, the current SyslogServer block is ignored.

5.6.2 Attribute "Enabled"

This attribute is a Boolean value ("Yes" or "No") that defines if the current SyslogServer block has to be ignored or not.

If this attribute is not set, the current SyslogServer block is ignored.

5.6.3 Attribute "Port"

This attribute is an Integer value that defines the destination port used by the remote Syslog server.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultPort" is used.

5.6.4 Attribute "Protocol"

This attribute is a String value that defines the layer protocol used by Syslog protocol. The only supported value for now is "udp".

If this attribute is not set in the configuration file, then the value set for attribute "DefaultProtocol" is used.

5.6.5 Attribute "MaxLength"

This attribute is an Integer value that defines the maximum length of a Syslog message. Be careful with this value because the RFC 3164 (the Syslog RFC) clearly indicates that the maximum length of a Syslog frame should be lower than 1024.

If this attribute is not set in the configuration file, then the value set for attribute "DefaultMaxLength" is used.

5.6.6 Attribute "EndOfMessage"

This attribute is a String value that defines special characters that can be added at the end of each Syslog message. The only supported values are:

- "": this empty sequence will add nothing at the end of the Syslog message.
- "CR": this will add a carriage return code (ASCII value \r or 0x0d) at the end of the Syslog message.
- "LF": this will add a line feed code (ASCII value \n or 0x0a) at the end of the Syslog message.
- "CRLF": this will add a carriage return code followed by a line feed code (ASCII values \r\n or 0x0d 0x0a) at the end of the Syslog message.

If this value is not set in the configuration file, then the value set for attribute "DefaultEndOfMessage" is used.

5.7 The <Heartbeat> parameter block

This block contains the parameters used to define the behavior of the WinAgentLog heartbeat feature.

5.7.1 Attribute "TimeoutMs"

This attribute is an Integer value that defines the delay in milliseconds before the WinAgentLog service send a heartbeat message.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "60 000" (one heartbeat message every minute).

5.7.2 Attribute "Enabled"

This attribute is a Boolean value that defines if WinAgentLog service is allowed to send heartbeat messages.

If this value is not set in the configuration file, then the default "hard coded" value in the program is set to "true" (WinAgentLog service is allowed to send heartbeat messages).

5.7.3 Attribute "Facility"

This attribute is an Integer value that defines the Syslog facility used by heartbeat messages.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "23" (Syslog local7 facility).

5.7.4 Attribute "Severity"

This attribute is a, Integer value that defines the Syslog severity used by heartbeat messages.

If this attribute is not set in the configuration file, then the default "hard coded" value in the program is set to "6" (Syslog informational severity).

Chapter 6
Formatting log messages

Preparing a Syslog message and sending it to a Syslog server follows 3 steps:

- Removing unwanted characters.
- Applying a format.
- Truncating and translating.

6.1 Removing unwanted characters

The first thing which is modified in an EventLog message is non ASCII characters. Any carriage return, line feed or tabulation characters are replaced.

- Carriage return character (ASCII code CR or ASCII value 0x0d) is replaced by the string "\r".
- Line feed character (ASCII code LF or ASCII value 0x0a) is replaced with "\n" string.
- Tabulation character (ASCII code TAB or ASCII value 0x09) is replaced with "\t" string.

6.2 Applying a format

When forwarding an EventLog message to a Syslog server, it is possible to define which information will be written into the message. This is done by specifying a message format with special values in it. The message format to use is specified by the attributes set for "Format" or "DefaultFormat" in the configuration file.

When creating the message, the WinAgentLog service will replace some special values by their real meaning as found in the EventLog message.

The following table shows existing special values and the value they are replaced by.

Special value name	Replaced by
$PROVIDER	Replaced by the name of the EventLog source.
$EVENTID_LOW	Replaced by the low part of the EventLog message identifier.
$EVENTID_HIGH	Replaced by the high part of the EventLog message identifier.
$EVENTID_FULL	Replaced by the complete EventLog message identifier.
$LEVEL_ID	Replaced by the severity code number of the EventLog message.
$LEVEL_STR	Replaced by the severity code string of the EventLog message.
$RECORD_NUMBER	Replaced by the record number of the record in EventLog location database.
$PROCESS_ID	Replaced by the process identifier of the process that generated the EventLog message.
$THREAD_ID	Replaced by the thread identifier of the process that generated the EventLog message.
$CHANNEL	Replaced by the name of the EventLog location.
$COMPUTER	Replaced by the name of the host where the EventLog message has been generated.
$MESSAGE	Replaced by the EventLog message.
$MSG_ID	Replaced by the message number. This number is generated by WinAgentLog service. Its value is 0 when starting and incremented for each new EventLog message.
$FACILITY_ID	Replaced by the number value of the EventLog identifier facility part.
$FACILITY_STR	Replaced by the string value of the EventLog identifier facility part.
$SEVERITY_ID	Replaced by the number value of the EventLog identifier severity part.
$SEVERITY_STR	Replaced by the string value of the EventLog identifier severity part.
$TIME_CREATED	Replaced by the time at which this entry was received by the EventLog subsystem and written into the EventLog location database. The format of the date is specified by the attributes set for "DateFormat" or "DefaultDateFormat" in the

	configuration file.
$USER_SID	Replaced by the SID (Security Identifier) of the user that generated the EventLog message. This information may be not available if it has no meaning in the EventLog context.
$USER_NAME	Replaced by the real name of the user that generated the EventLog message. This information may be not available if it has no meaning in the process context.

For example, with the following format string:

```
Format: "MSG_ID='$MSG_ID', RECORD_NUMBER='$RECORD_NUMBER', TIME_CREATED='$TIME_CREATED', FACILITY_ID='$FACILITY_ID',
FACILITY_STR='$FACILITY_STR', SEVERITY_ID='$SEVERITY_ID', SEVERITY_STR='$SEVERITY_STR', COMPUTER='$COMPUTER',
CHANNEL='$CHANNEL', PROVIDER='$PROVIDER', EVENTID_LOW='$EVENTID_LOW', EVENTID_HIGH='$EVENTID_HIGH',
EVENTID_FULL='$EVENTID_FULL', LEVEL_ID='$LEVEL_ID', LEVEL_STR='$LEVEL_STR', PROCESS_ID='$PROCESS_ID', THREAD_ID='$THREAD_ID',
USER_SID='$USER_SID', USER_NAME='$USER_NAME', MESSAGE='$MESSAGE'"
```

The following messages can be generated:

```
Message 1: "Dec 22 17:18:20 192.168.1.100 MSG_ID='8', RECORD_NUMBER='98529', TIME_CREATED='22/12/2010 16:18:19',
FACILITY_ID='16', FACILITY_STR='local 0', SEVERITY_ID='6', SEVERITY_STR='Information', COMPUTER='lumiere', CHANNEL='System',
PROVIDER='Service Control Manager', EVENTID_LOW='7036', EVENTID_HIGH='16384', EVENTID_FULL='1073748860', LEVEL_ID='4',
LEVEL_STR='Information', PROCESS_ID='784', THREAD_ID='392', USER_SID='', USER_NAME='', MESSAGE='Le service Service de rapport
d'erreurs Windows est entré dans l'état : arrêté.'"


Message 2: "Dec 22 17:20:42 192.168.1.100 MSG_ID='12', RECORD_NUMBER='98530', TIME_CREATED='22/12/2010 16:20:41',
FACILITY_ID='16', FACILITY_STR='local 0', SEVERITY_ID='6', SEVERITY_STR='Information', COMPUTER='lumiere', CHANNEL='System',
PROVIDER='Microsoft-Windows-Winlogon', EVENTID_LOW='7001', EVENTID_HIGH='0', EVENTID_FULL='7001', LEVEL_ID='4',
LEVEL_STR='Information', PROCESS_ID='3348', THREAD_ID='3104', USER_SID='S-1-5-18', USER_NAME='AUTORITE NT\Système',
MESSAGE='Notification d'ouverture de session utilisateur pour le Programme d'amélioration de l'expérience utilisateur' "
```

6.3 Truncating and translating

The third step is to truncate the length of the message to the value specified by the attribute "MaxLength" or "DefaultMaxLength".

And finally, the message is translated into UTF-8 character set before forwarding it to a Syslog server.

Chapter 7
Debugging WinAgentLog service

The WinAgentLog service writes debug messages to help the Circitor in case of problem or bug. The program uses the WinLog DLL to write log messages.

- This debug file is always named "WinLog_WinAgentLog.log" and its default location is in "c:\".
- The environment variable "WINLOG_FEATURE_LEVEL" can be set to define which log level will be written to the log file. If set to 0, only few number of logs are generated. If set to 4 294 967 295, all possible log messages are written to the log file. You can set any of the 32 bits to enable level 1 to level 32, level 0 is always set.
- The environment variable "WINLOG_FILE_FOLDER" can be set to change the folder where to generate log files. If not set, the default value is "c:\"
- The environment variable "WINLOG_FILE_SIZEMAX" can be set to define the maximum size in bytes of a log file before rotating it. If not set, the default value is 1 000 000 bytes.
- The environment variable "WINLOG_FILE_HISTORYSIZE" can be set to define the maximum number of history log file. If not set, the default value is 10 files.
- If the environment variable "WINLOG_FILE_ENABLED" is set to 0, the debug log feature is totally disabled.

End of reference manual