

The Sentry-go Scripting Wizard

Scripting just got easier!

Last Updated Saturday, 17 April 2010

© 3Ds (UK) Limited
<http://www.Sentry-go.com>

Be Proactive, Not Reactive!



Table of Contents

Symbols	2
Scripting, the easy way!	2
How It Works	3
Version 2.x Enhancements.....	3
Installing.....	3
Running the Wizard from within Sentry-go	4
Choosing a Template or Script File	5
Script Information	6
Generated Script Name.....	7
Entering Parameter Information	8
Generating the File or Script.....	9
Review the File or Script.....	10
Writing Script Templates using the Scripting Wizard API	12
Special Considerations	29
Script Compilation Errors.....	29
Custom Monitoring, Utilities & Wizards	30
About 3Ds (UK) Limited.....	30



Symbols

Thank you for choosing Sentry-go® as your monitoring solution for Windows. In this guide, the following symbols are used to denote specific items ...

-  Important information which should be noted – it may affect what you are trying to do.
-  Additional information relating to the operation being described is shown.

Scripting, the easy way!

Over the past few years, scripting has grown ever more popular, especially as related tools and interfaces have also been improved. Scripts have a number of different uses & can be an extremely effective way of performing one or more tasks, for example ...

- To check something is working
- To perform an action in order to resolve a fault
- Help alert you when something goes wrong

More often than not, Sentry-go monitoring solutions will more easily perform these tasks and with greater flexibility, but even then, there may still be times when you wish to have more programmatic control over the logic that's being run. This is where the ability to access scripts comes in.

Traditionally, scripting was reserved for those who either had some programming knowledge & could therefore write & debug scripts to perform the task, or those who had time to modify someone else's scripts. While these are still very valid options today, it would also put many people off - even when the actual task they wanted to perform was quite easy to achieve with just a few lines of code.

The Sentry-go Scripting Wizard is designed to help remedy this by targeting two types of user ...

- It allows those with little or no programming/scripting knowledge to access pre-written scripts & simply enter information in order to customise them when prompted by the Wizard.
- It allows those with scripting experience to provide generic logic (the template files) without needing to know the specific details of the task etc. These can be prompted later when the end user comes to use/generate the final script.

In addition, the Sentry-go Scripting Library (<http://www.Sentry-go.com/sentry-go-scripting-library.aspx>) has many downloadable script templates for you to use or modify as required.

The sections below show how to both run the Wizard to create a script from an existing template – which can then be defined to Sentry-go as either a monitoring, auto-response or alerting action, or write the template itself, allowing the end user to be prompted for the required information etc.



How It Works

As an end user, the aim of the Sentry-go Scripting Wizard is very simple - to generate a script that will allow you to run the task you need to perform without needing to write any scripting code yourself.

It does this by accessing a template – a file which has already been written and contains the underlying syntax and logic. Templates differ, however, from standard scripts because they typically contain one or “Wizard tags” (“tags”), that request information from the end user which is then placed into the file. This could be, for example, the name of a server or filename etc.

A tag is simply a string of characters, written within the script that tells the Scripting Wizard to do something – typically to request one or more pieces of information that will then be used within the script. Once the information is gathered, the Wizard generates the final script, removing the tags automatically and replacing them with the answers needed to create the file that you will ultimately run.

- If you are using the Scripting Wizard as an end user, you simply select the template that performs the task you wish to run & answer the questions presented to you. Then watch as the script is generated!
- If you're a developer, you have the option of including tags within your own scripts (to make them templates) in order to have the Wizard customise your own logic. This in turn, allows end-users to tailor *your* scripts for *their* own use.

Version 2.x Enhancements

The Scripting Wizard was introduced originally as part of Sentry-go Plus! where it allowed you to generate one or more alerting scripts. In this release, the entire monitoring solution benefits from many functional improvements such as these ...

- Scripts are no longer restricted to those supplied by the Wizard.
- Templates now provide alerting, monitoring & auto-response-style scripting.
- An API now allows developers to integrate their own scripts with the Wizard.
- The user interface now allows for a far greater number of tags to be supported.
- Many new scripts have been added & can be downloaded from the scripting library.
- Scripts generated by the Wizard can be used with the Sentry-go monitoring solution, including alerts, responses & the Scripted monitoring component, where installed.
- Scripts can also support calling parameter & return value information.

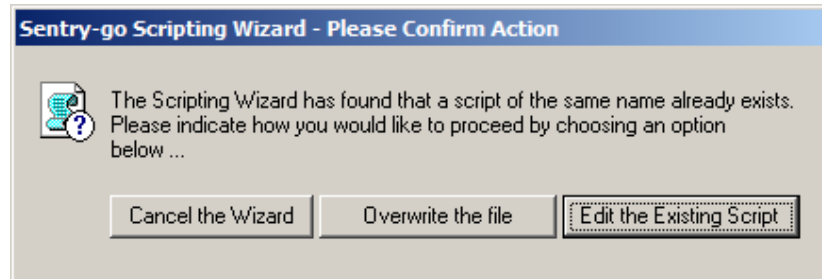
Installing

The Sentry-go Scripting Wizard comes via Electronic Software Download (ESD) as a single ZIP file. It is also available as part of other Sentry-go solution downloads. Simply expand this file into a directory & run Wizard's executable application.



Running the Wizard from within Sentry-go

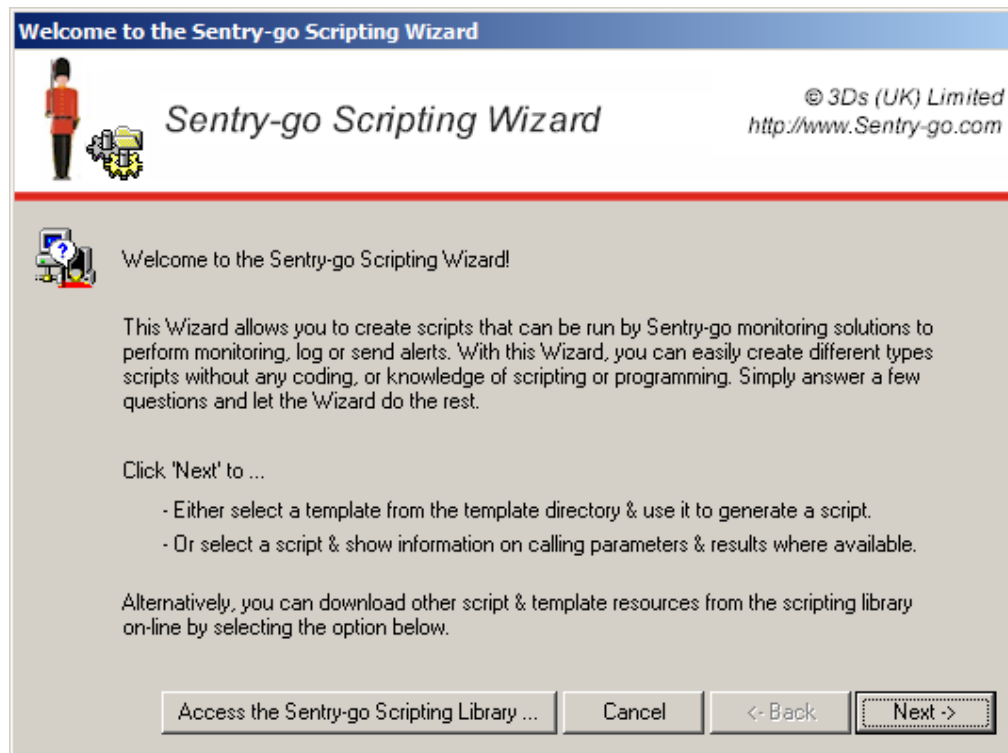
To run the Wizard, simply run the utility's executable file or access it from within the Sentry-go Client Console. If a script has already been defined & exists, it will prompt you as follows ...



From here you can cancel the Wizard, overwrite the file & continue to generate a new script based on a chosen template or edit the script that currently exists. When you edit the script, the Wizard also checks to see if it supports parameter & result help tags. If it does you also have the option of finding out if any parameters are required and the codes/results the script might return.

For more information, see "Review the File or Script" below.

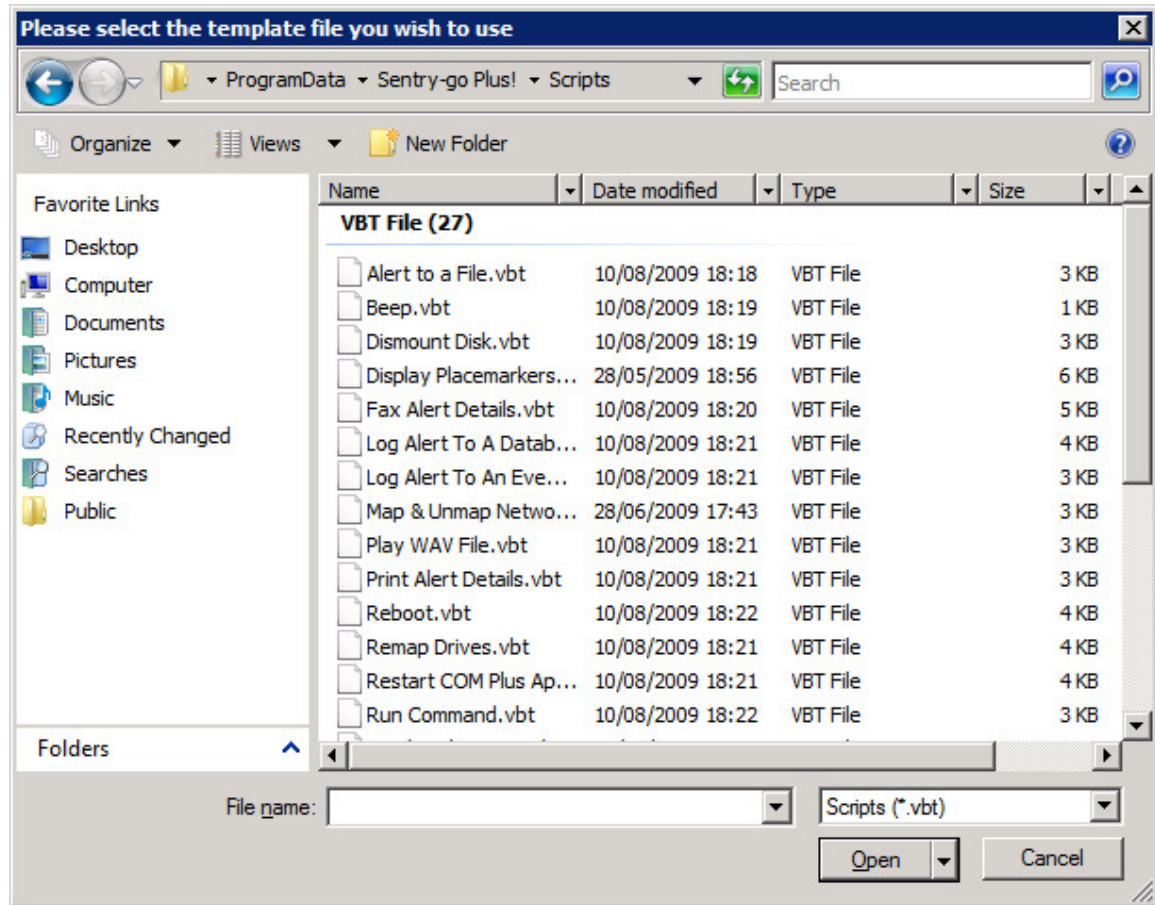
If you choose to continue, or no existing script is found, an introductory window, similar to this, will initially be displayed ...




From here, click Next to continue.

Choosing a Template or Script File

The next window allows you to select the template you wish to run through the Wizard. The template you select will be determined by the task you wish to perform ...



Simply select the template that matches your requirement & click "Open". Alternatively, you can navigate to a different location and choose a template from there.

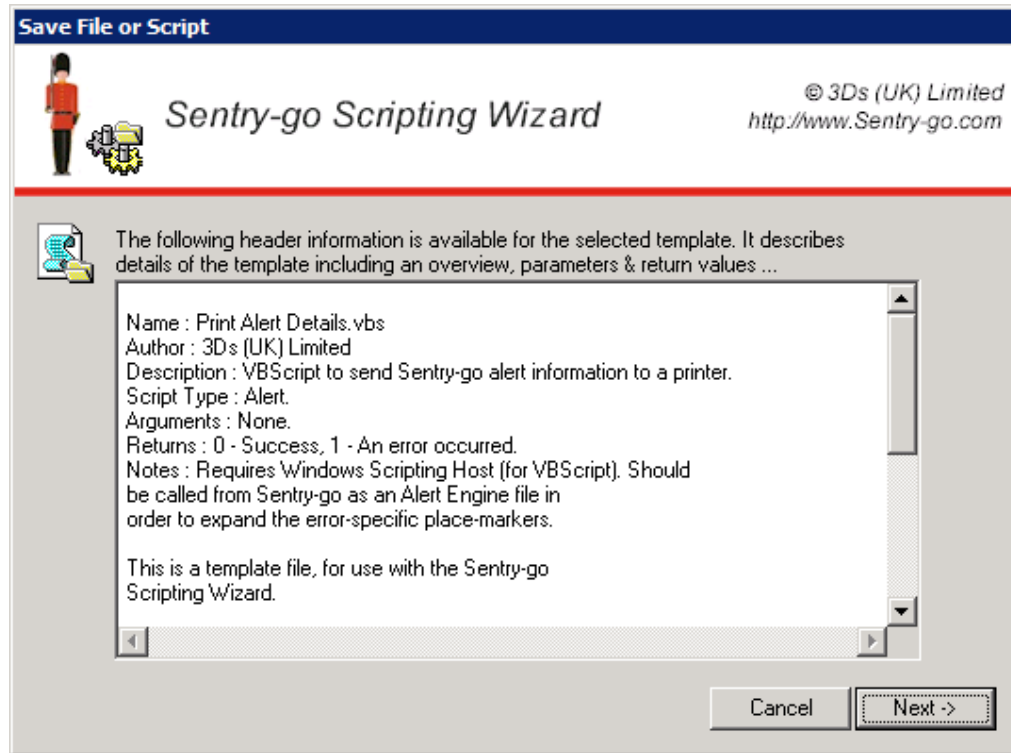
 If you cannot find the appropriate script, visit the Sentry-go Scripting Library on-line. This provides a number of scripts & files which are available for download. Once downloaded, navigate to the new file and click "Open".

If you open a script file, it is unlikely that any Wizard parameters will be found. However, if help is available for the command line parameters and/or the return code or results returned, the Wizard will still display these to help you define the script to Sentry-go.

Many Sentry-go scripts contain the above command line &/or results help.

Script Information

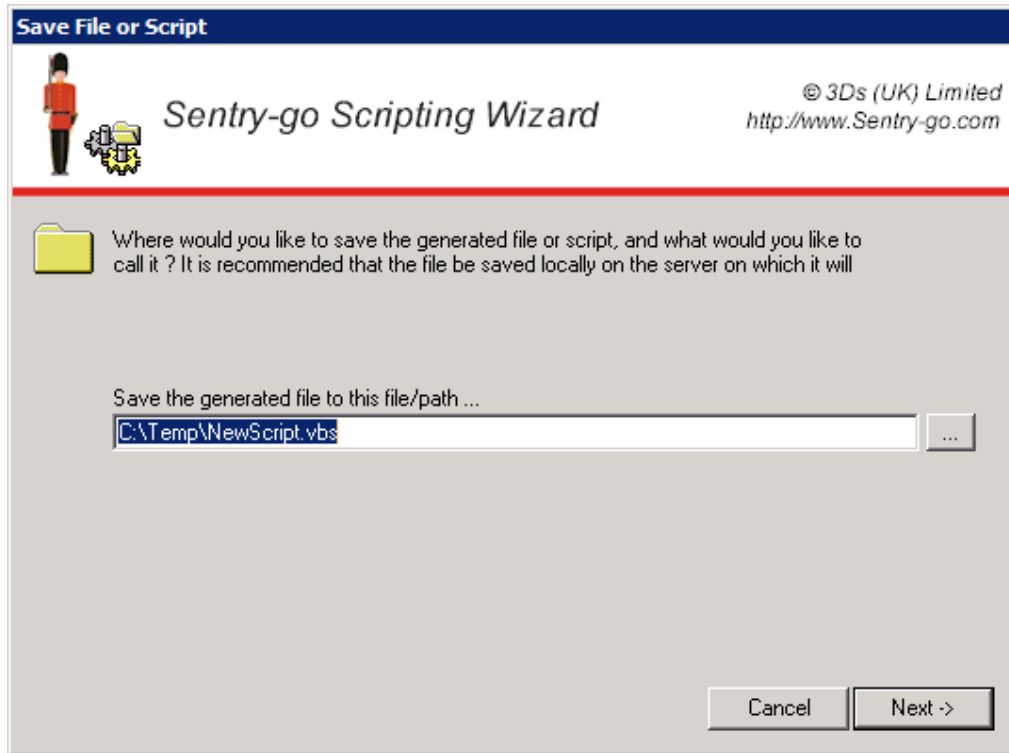
If scripting information is available, the next window shows this – for example, its name, description, parameters, return type, compilation results etc.



Click "Next" to continue.

Generated Script Name

Unless specified before, the following window will be displayed, allowing you to select the location & name of the script or file the Wizard will generate. You can either enter the path/name, or click “...” to select the file from a list.



Once complete, click “Next” to continue.

Entering Parameter Information

The script will now be processed by the Wizard. Before it can generate the final script, the Wizard may need to request more information – determined by the contents of the script & in particular any tags found within it..

Information is requested using a simple interface, designed to help you enter the correct answer without the need for much technical knowledge ...



Information windows are split into a number of specific parts.

The Question

The question itself indicates what information is required. This information will then be placed into the final script when it is generated.

The Answer

The edit field allows you to enter the answer to the question – this may be text-based, selection from a list, the selection of an option – e.g. yes/no, or specific information such as an SMTP connection etc.

Additional Help

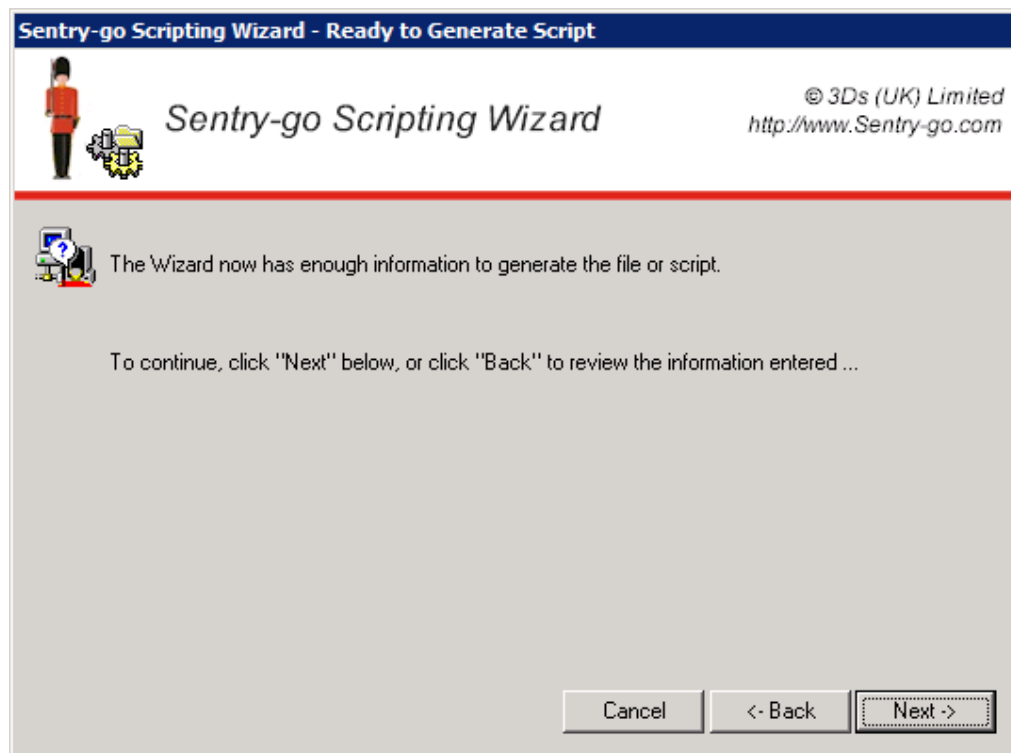
In many cases, additional help will be displayed at the bottom of the window, giving extra advice as to the details required to be entered or the choice to be made.

Selections

In some cases, you can choose or select the answer. This may be from a dropdown list, or by clicking the “...” button to display additional selection information.

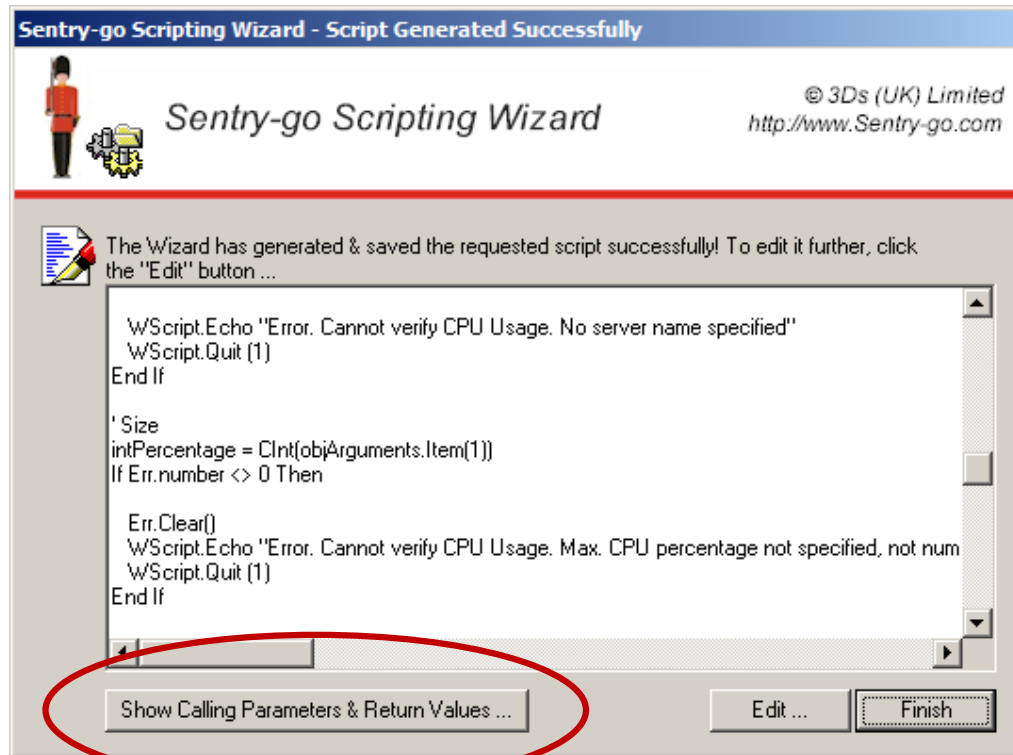
Generating the File or Script

Once the Wizard has collected all the information it needs to generate the script, the following window will be displayed. Click “Back” to review the answers already entered or “Next” to continue to generate the final file.



Review the File or Script

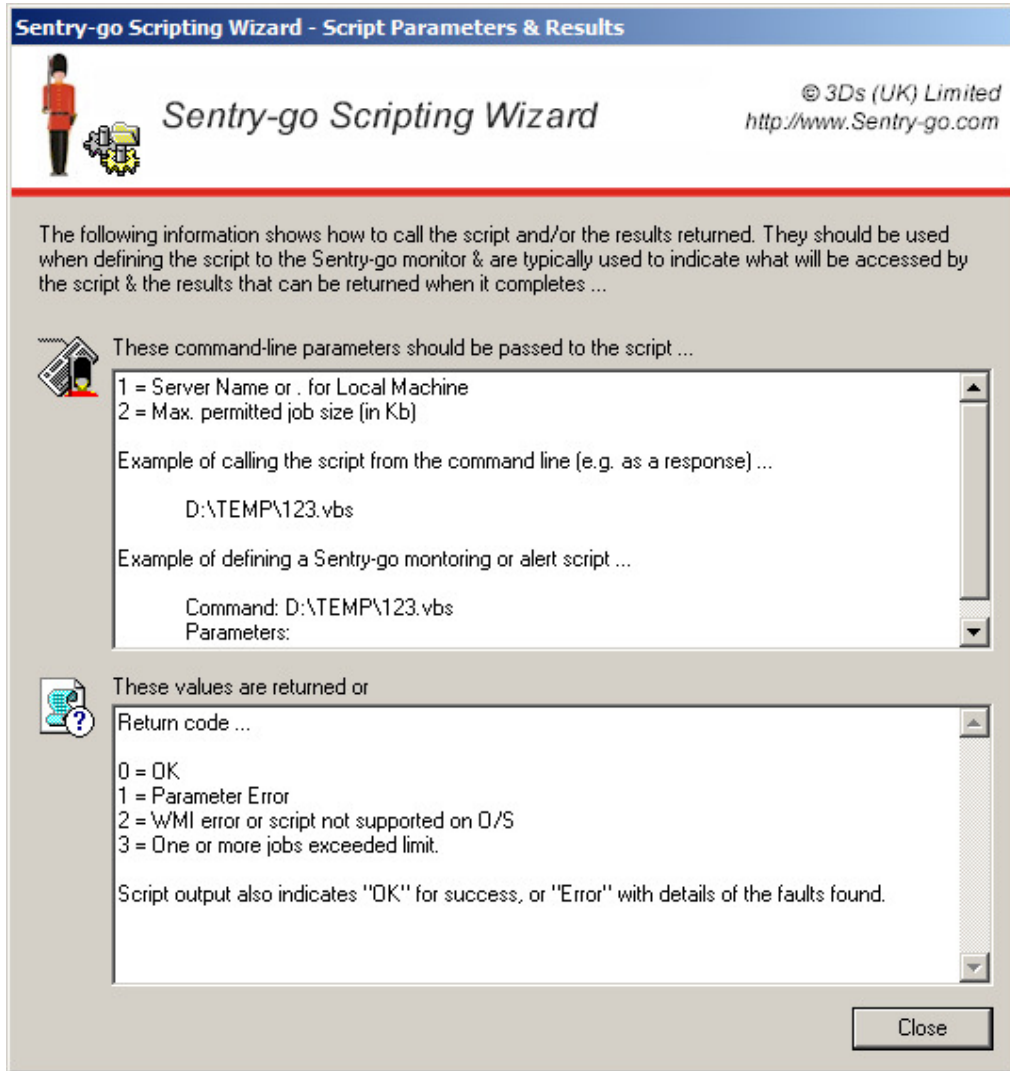
When the Wizard has generated the script, the result will be displayed to you. If “TODOs” are found, you will be prompted to edit the script further. You can do this by clicking the “Edit” button from this window.



Show Calling Parameters & Return Values

If additional help is available to show how to call the script and/or the values it returns, the “Show Calling Parameters & Return Values” button will be available.

Click it to see additional help on running & defining the script to the Sentry-go monitor.



The top window gives information of the parameters that the script expects when it is called. It also shows you examples of how a call would be made.

The lower window gives information on the return codes and any output the script might make.

When complete click Close, then "Finish" from the main window.



Writing Script Templates using the Scripting Wizard API

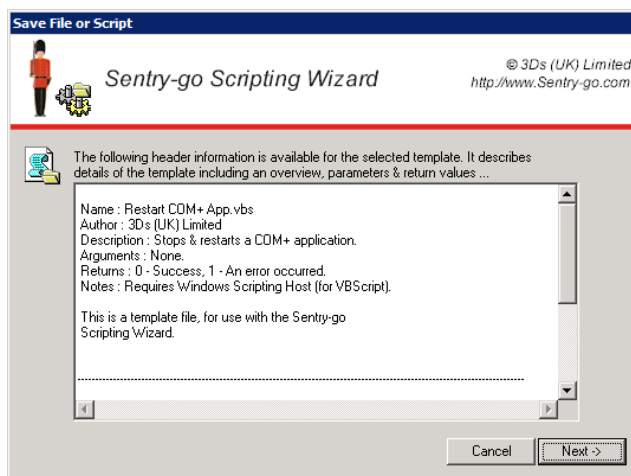
In addition to running the Wizard, you can also create new templates. Templates allow the wizard to prompt the user for additional information as well as generate the final script.

The notes that follow are aimed at script developers to help enable them write scripts for use for the Wizard.

Language Support

The Wizard supports any text-based scripting engine so you can use any scripting language that is available to you – including batch files. In addition, Sentry-go alert files can also be generated using the Wizard.

The Script Header



The Wizard will automatically display header information to the end user when they select the template.

The Wizard does this by searching the script for "****" or "####". When found, the information that follows is formatted until the next "****" or "####" is found. If these characters are not located, the script is assumed not to have a header and the details are suppressed.

Below is an example of a typical header file. As you can see, it tells the end user all they need to do about parameters & return codes etc.

```
' *****  
'  
' Name       : Print Alert Details.vbs  
' Author     : 3Ds (UK) Limited  
' Description : VBScript to send Sentry-go alert information to a printer.  
' Script Type : Alert.  
' Arguments  : None.  
' Returns    : 0 - Success, 1 - An error occurred.  
' Notes      : Requires Windows Scripting Host (for VBScript). Should  
'              be called from Sentry-go as an Alert Engine file in  
'              order to expand the error-specific place-markers.  
'  
'              This is a template file, for use with the Sentry-go  
'              Scripting Wizard.  
'  
' *****
```

In addition, the header can optionally include parameter & return value information as shown below.



Command-line Parameter & Return Value Support

Within the header, the following options can also be included. These tell the Wizard how the script is called & the return values that might be expected. It is used to display the information from the "Edit Script" window once the Wizard has completed.

- WIZARD:PARAMS=<Text>|<Text>

Include this line within the header block to indicate the parameters expected by the script when it is run. All text should be included on the same line. However, you can use the "|" character to indicate that a new line should be displayed by the wizard.

```
WIZARD:PARAMS=1 = Server Name|2 = Name of Windows Service
WIZARD:PARAMS=No parameters are used by the script
```

- WIZARD:PARAMEXAMPLE=<Example Text>

Include this line within the header block to show an example of calling the script (without the script name itself).

```
WIZARD:PARAMEXAMPLE=YourServerName Alerter
WIZARD:PARAMEXAMPLE=
```

- WIZARD:RESULTS=<Results or Return Codes>

Include this line within the header block to indicate how the script will return a result. This may be numeric, text (i.e. text output to the screen when run) or both. All text should be included on the same line. However, you can use the "|" character to indicate that a new line should be displayed by the wizard.

```
WIZARD:RESULTS=Return code ...||0 = OK|1 = A WMI error occurred or the script is not supported|2 =
An error prevented the script from completing successfully.||If an error occurs, a message will be output
giving details of the fault encountered.
```

Wizard "Tag" Syntax

The Wizard supports a number of its own commands which it searches for when compiling the script or file. These are always written in the same format and take the place of the answer that will ultimately be entered into the script.

In order to ensure the syntax of the final script is correct, it is important to ensure answers returned by a particular tag is correctly entered into the file. The tag takes the place of the answer as it will appear in the script. Once removed, only the answer will be left, so if this is a string, it should be in quotation marks.

As a general rule ...

- If the script expects a string, enter the whole tag in the quotation marks that would be normally used by the script.

```
strPrinter = "<PARAM:Type=Text;Question=Print to this printer port;Default=LPT1:>"
```

- If the script expects a non-string value such as a number or Boolean value, enter the tag into the script without quotation marks.

```
bForce = <PARAM:Type=Option;Question=Would you like to force a reboot ?;Data=Yes=True,No=False>
```

The syntax of each tag is dependent on the command itself and is given in detail below. But in general it follows this format ...

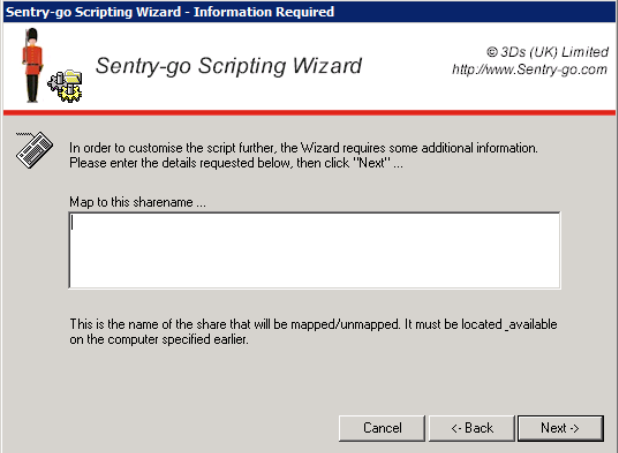
```
<PARAM:Type=[Type];Question=[Question Text];Help=[Help Text]; [Other Parameters]>
```



Type=Text

Requests a text-based answer from the end user.

An answer is required; an error will be displayed if no entry is made.



Parameters ...

Question=[Text]

This is the question text that will be displayed.

Help=[Text]

Optional. Text that helps the user determine what should be entered.

Default=[Answer]

Optional. The default answer, displayed in the answer field when the question is first displayed. Can be overwritten by the user but can help them understand the information required, or provide a default answer for your script.

MaxLength=[Value]

Optional. The maximum permitted length of the answer. An error is displayed if this length is exceeded.

NumericOnly

Optional. Specify this option if the value entered must be a numeric number such as a port no.

Additional Information ...

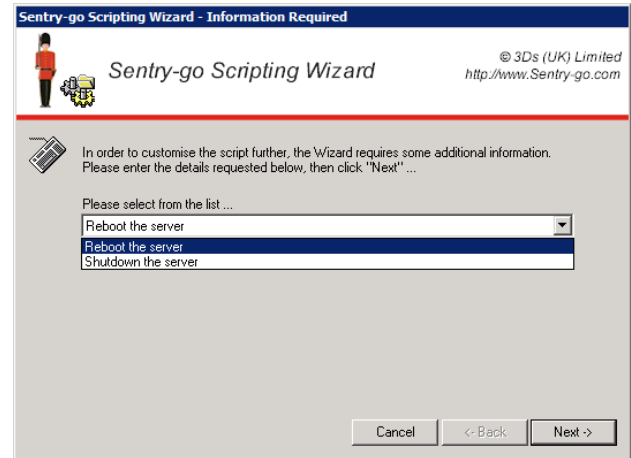
Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

Type=List

Allows the user to select an entry from one of those listed in the dropdown list.

Optionally, they can also enter a free-text value.



Parameters ...

Question=[Text]
Help=[Text]
Data=[Listed Items]

This is the question text that will be displayed.
Optional. Text that helps the user determine what should be selected or entered.
The data field is a list of one or more entries that will be displayed in the dropdown list. It can be in one of these formats ...

- Data=First Value,Second Value,Third Value, ...

In this case, the selected value (e.g. "First Value") is returned as the answer within the script.

- Data=First Value=1,Second Value=2,Third Value=3, ...

In this case, the selected value (e.g. "First Value") is translated when the Wizard returns the result. The value returned is the value matching the selected option – e.g. First Value would return the number 1, Second Value would return 2 and so on. Values can either be represented as numbers or string values.

In this case, if the user enters their own value – i.e. one that's not on the list, the entered value will become the result.

NoEdit

Optional. Specify this option if the value must be selected from one of those shown in the list. In other words, the user cannot enter their own value in response to the question.

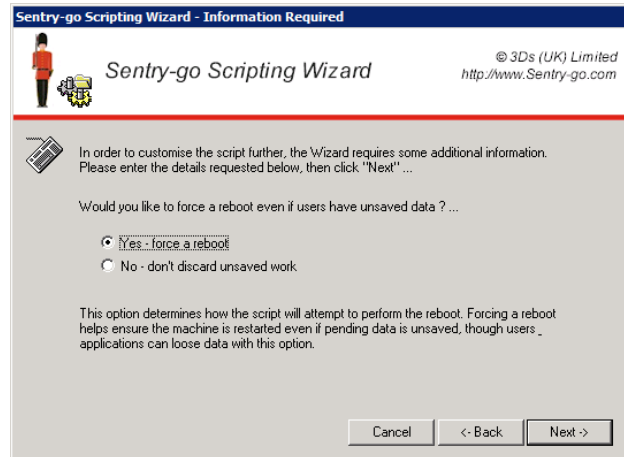
Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

Type=Option

Allows the user to select from two available options– e.g. Yes/No etc.



Parameters ...

Question=[Text]

This is the question text that will be displayed.

Help=[Text]

Optional. Text that helps the user determine what should be selected.

Data=[Listed Items]

The data field provides the two options that will be displayed. It can be in one of these formats ...

- Data=Yes,No

In this case, the selected value (e.g. “Yes”) is returned as the answer within the script.

- Data=Yes – show me the result=True,No – don’t display details=False

In this case, the selected value (e.g. “Yes – show me the result”) is translated when the Wizard returns the result. The value returned is the value matching the selected option – e.g. True.

Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

Type=MessageTitle Type=MessageText

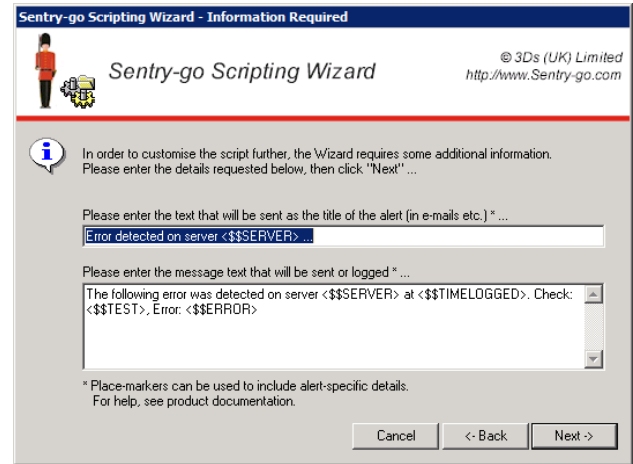
Both these options will cause the user to be prompted for error details as a single window.

To include the appropriate text – i.e. the title or the text, simply specify the appropriate tag within the template.

Default values are displayed automatically. However, these can be overwritten later as described below.

Please enter the text that will be sent as the title of the alert (in e-mails etc.) * ...

Enter the text that will become the title of the e-mail.



You can include any text, including Sentry-go place-markers such as ...

- <\$\$SERVER> The name of the local server
- <\$\$ERROR> The error message (from a log file) or the error string raised by the monitor
- <\$\$TEST> or <\$\$TITLE> The name or title of the test that was performed and ultimately raised the alert
- <\$\$TIMELOGGED> The time the error was raised in the format dd/mm/yy hh:mm.ss
- <\$\$TIMENOW> The current time in the format dd/mm/yy hh:mm.ss
- <\$\$OSNAME> The type of the server's Operating System - e.g. Windows 2000, Windows 2003, Windows XP etc.
- <\$\$SERVICEPACK> The latest service pack string, if applicable

For a full list of available place-markers (some of which are dependent on the monitor installed), please refer to your Sentry-go documentation.

Please enter the message text that will be sent or logged * ...

Enter the text that will become the text that is either sent as the e-mail, or logged to the server etc.

- You can include any text, including delimiters (e.g. a for a CSV file), heading or literal text etc.
- Place-markers can be used to include error specific information.
- For VBScripts, formatting characters can be included – e.g. “Chr(13) & Chr(10)” etc.
- Built-in functions can be used to format the message text – e.g. <\$\$ERROR,100> to limit the amount of text to the first 100 characters. This is particularly useful where you want the message text to be fixed in length such as an SMS message.

Parameters ...

None

Additional Information ...

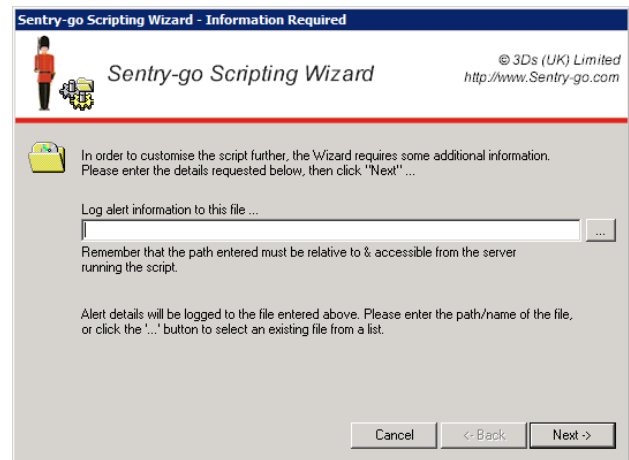
Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=File

Asks the user to enter a valid path & filename that will be used within the script.

The user can either enter a path, or select it using the “...” button.



Parameters ...

Question=[Text]
Default=[Answer]

This is the question text that will be displayed.

Optional. The default answer, displayed in the answer field when the question is first displayed. Can be overwritten by the user but can help them understand the information required, or provide a default answer for your script.

Help=[Text]
GenerateName

Optional. Text that helps the user determine what should be selected.

Optional. If specified, the Wizard will automatically generate & display a temporary file name in the entry field. This can be used by the user or they can overwrite it with a new one.

Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

Type=Directory

Asks the user to enter a valid path that will be used within the script. The user can either enter a folder path/name, or select it using the “...” button.

Parameters ...

Question=[Text]	This is the question text that will be displayed.
Default=[Answer]	Optional. The default answer, displayed in the answer field when the question is first displayed. Can be overwritten by the user but can help them understand the information required, or provide a default answer for your script.
Help=[Text]	Optional. Text that helps the user determine what should be selected.

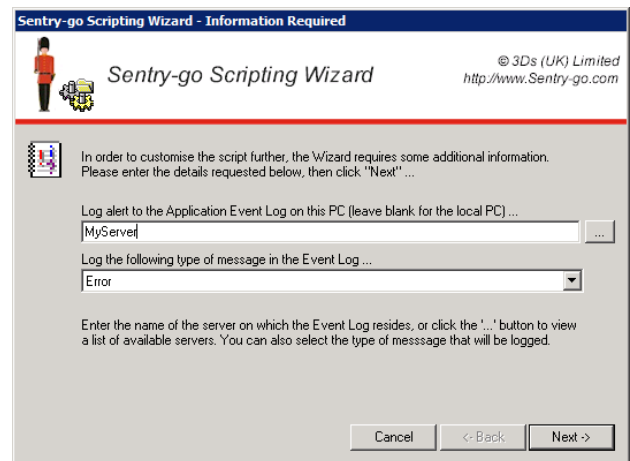
Additional Information ...

<i>Instancing</i>	<i>This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.</i>
-------------------	---

Type=EventLogPC Type=EventType

Both these options will cause the user to be prompted for Event Log details. In particular they will be asked ...

- The name of the PC where the Application Event Log resides
- The type of event to be logged (e.g. an error, warning, information event).



Parameters ...

None

Additional Information ...

<i>Instancing</i>	<i>Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.</i>
-------------------	---

Type=SMTPServer
Type=SMTPPort
Type=SMTPToEmail
Type=SMTPFromEmail

All these options will cause the user to be prompted for SMTP connection information.

In particular they will be asked to enter ...

- The name or IP address of the SMTP Server
- The port on which the server listens for requests (defaulted to 25)
- The e-mail address of the user that will receive the e-mail (the recipient)
- The e-mail address of the user that will send the e-mail

Parameters ...

None

Additional Information ...

Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

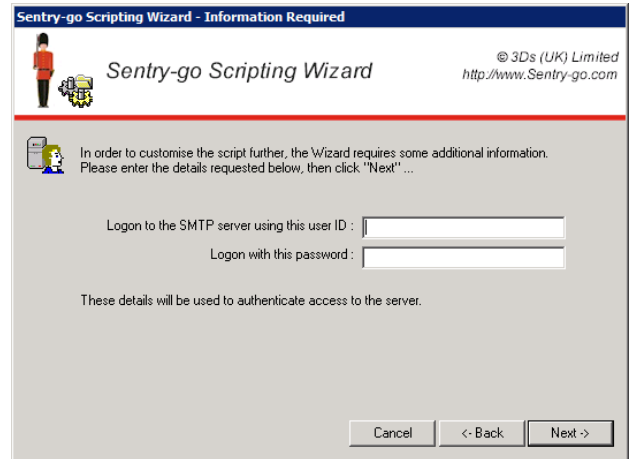
The screenshot shows a dialog box titled "Sentry-go Scripting Wizard - Information Required". The window has a blue title bar and a grey background. In the top right corner, it says "© 3Ds (UK) Limited" and "http://www.Sentry-go.com". The main text reads: "In order to customise the script further, the Wizard requires some additional information. Please enter the details requested below, then click 'Next' ...". Below this text are four input fields: "Name or IP-address of e-mail (SMTP) server:" with a dropdown arrow, "This server listens for requests on port:" with the value "25" entered, "Send message to this e-mail address:" with an empty field, and "Send message from this e-mail address:" with an empty field and a small example "(e.g. Someone@<YourDomain>)" below it. At the bottom right, there are three buttons: "Cancel", "<- Back", and "Next >".

Type=SMTPUser Type=SMTPPassword

Both these options will cause the user to be prompted for SMTP user information.

This is required only if your e-mail server uses authenticated SMTP.

If it does, specifying these tags will request the required information from the user.



Parameters ...

None

Additional Information ...

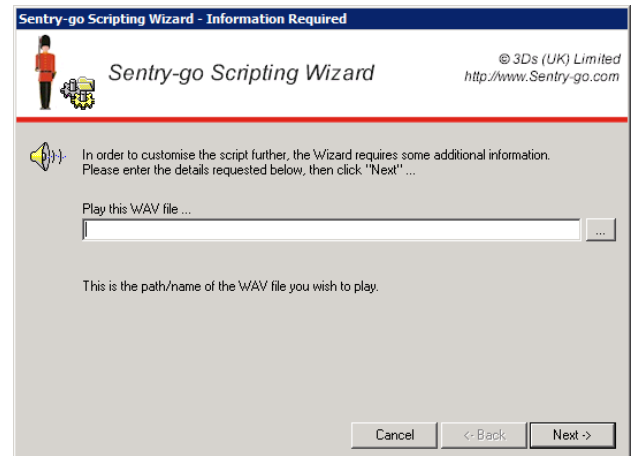
Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=WAVFile

Specifying this option will cause the user to be prompted for a WAV (sound) file, most typically when generating an alert template.

The user can enter the path/name of the WAV file or click “...” to locate it from the file system.



Parameters ...

Question=[Text]

Optional. Allows you to specify the question relating to the file being requested. If not specified, default text will be shown.

Help=[Text]

Optional. Allows you to specify additional help for the question being asked.

Default=[Answer]

Optional. The default answer, displayed in the answer field when the question is first displayed. Can be overwritten by the user but can help them understand the information required, or provide a default answer for your script.

Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.



Type=SysLogServer
Type=SysLogPort
Type=SysLogSeverity
Type=SysLogFacility

All these options will cause the user to be prompted for details of their SysLog server. In particular they will be asked to enter ...

- The name or IP address of the SysLog Server
- The port on which the server listens for requests
- The severity code of the logged message
- The facility the logging system should use

Parameters ...

None

Additional Information ...

Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=SMSHTTPServer
Type=SMSHTTPPort
Type=SMSHTTPConnectionMethod
Type= SMSHTTPPhoneNo
Type=SMSHTTPUser
Type=SMSHTTPPassword

All these options will cause the user to be prompted for information relating to the sending of an SMS gateway message using the HTTP (web) protocol.

Gateway providers typically allow the sending of SMS messages using either SMTP (e-mail) or HTTP (web) connectivity.

Sentry-go Scripting Wizard - Information Required

© 3Ds (UK) Limited
<http://www.Sentry-go.com>

Sentry-go Scripting Wizard

In order to customise the script further, the Wizard requires some additional information. Please enter the details requested below, then click "Next" ...

Name or IP-address of web (HTTP) server:

This server listens for requests on port:

Use this connection method:

Phone number to send SMS to:

Use this gateway user ID:

Use this gateway password:

Cancel <- Back Next ->

In particular, when any of these entries are found in the script, the user will be asked to enter ...

- The name or IP address of the gateway's web (HTTP) Server
- The port on which the server listens for requests
- The SMS phone number that the message is to be sent to
- The user's gateway user ID
- The user's gateway password

Parameters ...

None

Additional Information ...

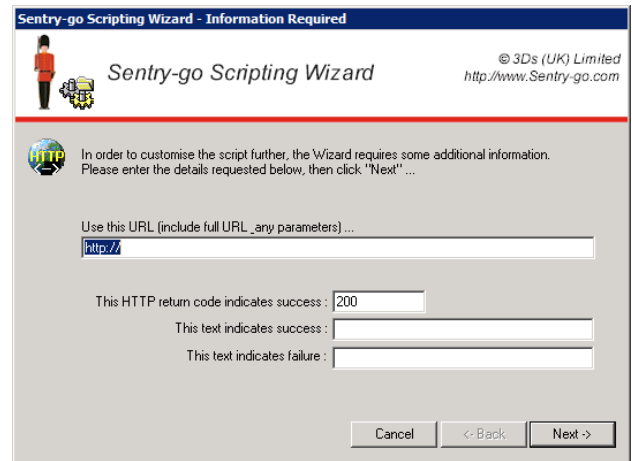
Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=SMSHTTPURL
Type=SMSHTTPResult
Type=SMSHTTPSuccessText
Type=SMSHTTPFailureText

All these options will cause the user to be prompted for additional information relating to the sending of an SMS gateway message using the HTTP (web) protocol.

Gateway providers typically allow the sending of SMS messages using either SMTP (e-mail) or HTTP (web) connectivity.



In particular, when any of these entries are found in the script, the user will be asked to enter ...

- The URL of the web server/page that you use to send the SMS via the gateway
- The HTTP result returned (typically 200) when successful
- Any text supplied in the resulting page that indicates that the message was sent successfully
- Any text supplied in the resulting page that indicates that the message could not be sent, or when an error occurs

The Wizard will prompt the user only once for this information, though the individual items of information can be included by specifying the tag at all appropriate locations within the template.

Parameters ...

None

Additional Information ...

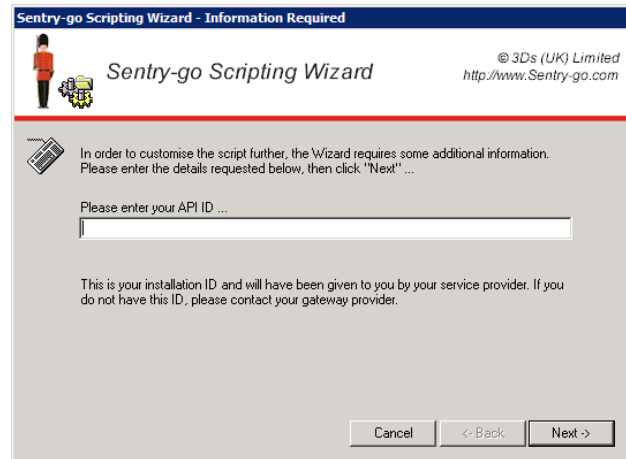
Instanting

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=SMSAPIID

If you are using the ClickATell gateway, you will be provided with an installation API ID.

You can use this tag to retrieve the API ID for use with a script sending an SMS message via HTTP or SMTP.



Parameters ...

None

Additional Information ...

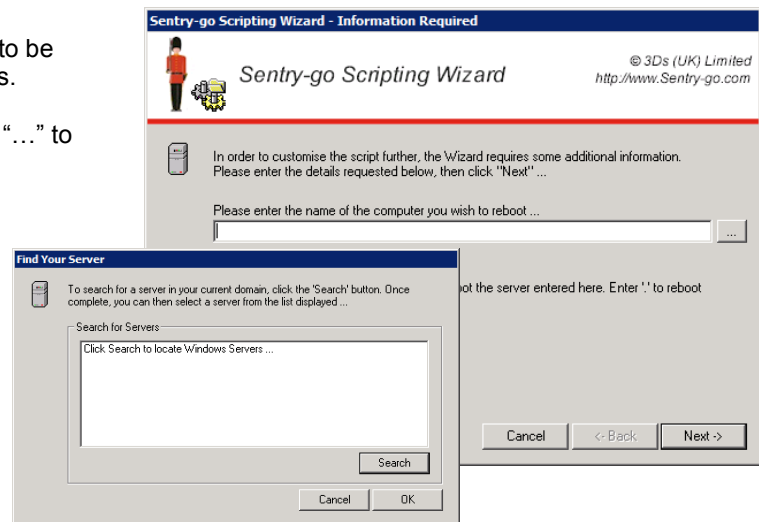
Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=Server

Specifying this option will cause the user to be prompted for a server name or IP address.

They can either enter this directly or click “...” to search for servers on their local domain.



Parameters ...

Question=[Text]

Optional. Allows you to specify the question relating to the server requested (e.g. the type of server required etc.) If not specified, default text will be shown.

Help=[Text]

Optional. Allows you to specify additional help for the question being asked.

Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

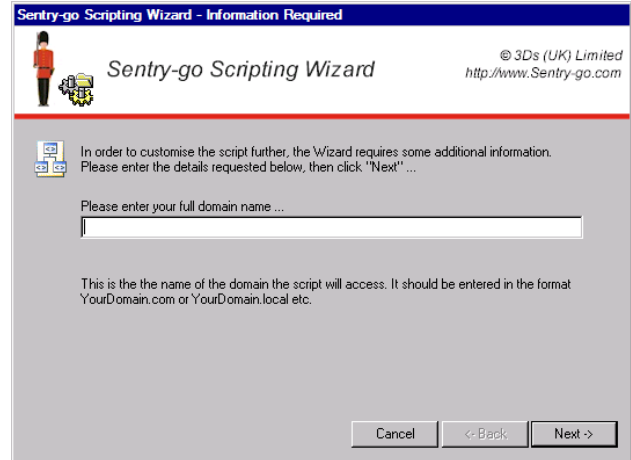


Type=Domain Type=DomainDC

Specify this option to request the user's domain name.

This should normally be entered in the format DomainName.com, DomainName.Local etc.

- When "Domain" is used, the name, as entered, will be inserted into the script.
- When "DomainDC" is specified, the name will be translated into a string in the format DC=<1st name>,DC=<2nd name> etc.



Parameters ...

Question=[Text]

Optional. Allows you to specify the question relating to the server requested (e.g. the type of server required etc.) If not specified, default text will be shown.

Help=[Text]

Optional. Allows you to specify additional help for the question being asked.

Additional Information ...

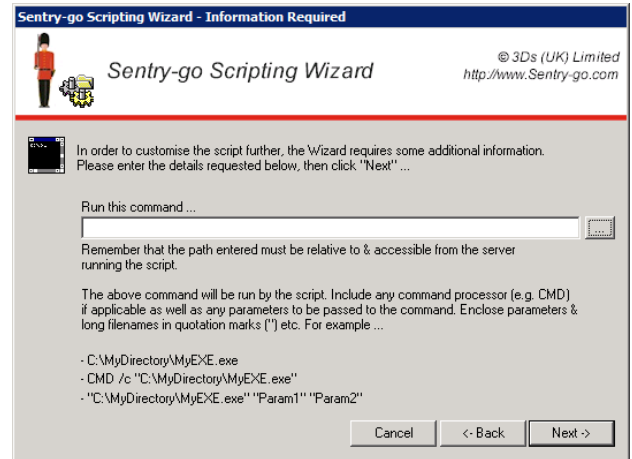
Instancing

Single. The first of either tag will request information from the user. Subsequent calls to either tag will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Type=RunCommand

Specifying this option will cause the user to be prompted for an executable command line. This may include optional parameters if required.

The user can enter the path/name of the executable, script or file themselves or click "..." to locate it from the file system.



Parameters ...

Question=[Text]

Optional. Allows you to specify the question relating to the command-line required. If not specified, default text will be shown.

Help=[Text]

Optional. Allows you to specify additional help for the question being asked.

Default=[Answer]

Optional. The default command-line (including any parameters), displayed in the answer field when the question is first displayed. Can be overwritten by the user but can help them understand the command syntax, or the default value you are expecting etc.

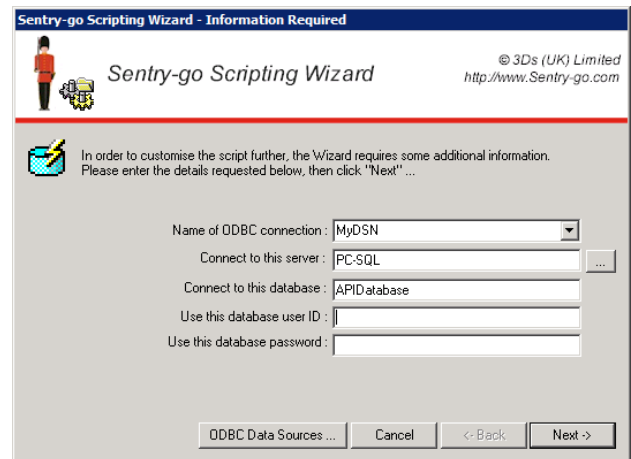
Additional Information ...

Instancing

This tag can be used multiple times within the script. Each occurrence will prompt the user for new information.

Type=ODBCDSN
Type=ODBCServer
Type=ODBCDatabase
Type=ODBCUser
Type=ODBCPassword

All these options will cause the user to be prompted for information relating to an ODBC connection – e.g. for connectivity via ADO etc.



In particular, when any of these entries are found in the script, the user will be asked to select or enter ...

- The ODBC data source (DSN) they wish to use
- Optional. The database server that the DSN should connect to.
- Optional. The name of the database the DSN should connect to.
- Optional. A database user ID that allows connectivity to the specified server/database.
- Optional. A password for the above user ID.

The Wizard will prompt the user only once for this information, though the individual items of information can be included by specifying the tag at all appropriate locations within the template.

Parameters ...

None

Additional Information ...

Instancing

Single. The first tag will request information from the user. Subsequent calls to the same or other tags in the group will utilise the information already given. Further information from the user will not be requested again. If another instance is required, use multiple instance tags – e.g. Type=Text instead.

Special Considerations

In some cases, where “instancing” is marked as single, the Wizard will only use one set of answers even if multiple tags are entered. In the unlikely event you wish to use a second instance of the information, use generic tags such as Type=Text to retrieve the information.

Script Compilation Errors

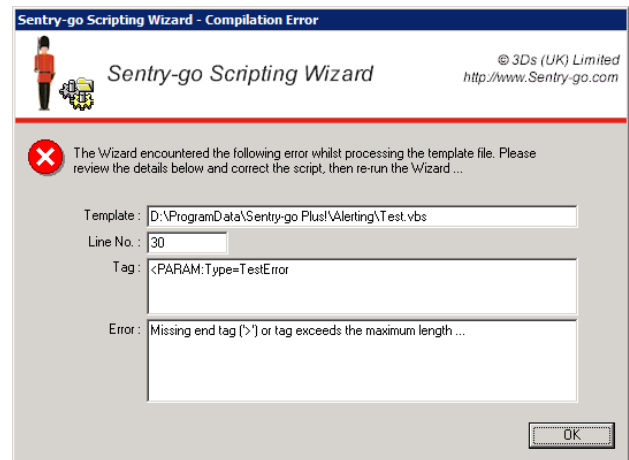
Although the Wizard does not run the script itself, it does perform a compilation of its own. Typically this is transparent to the end-user but in essence this step verifies the tags entered and determines which information (and which Windows) need to be requested from the end user.

The final script cannot be generated until the compilation process (which is normally extremely quick) has completed.

In the event the Wizard encounters a tag it does not understand – e.g. it is mistyped, or the end tag (“>”) is missing etc., a compilation error will be displayed.

The details shown will help determine the line of the script & the reason for the failure and is intended primarily as a debugging aid for the script developer.

If you are running the Wizard as an end-user and encounter this type of error, please refer it, where possible, to the original script developer.



Custom Monitoring, Utilities & Wizards

In addition to this and other Wizards, utilities & monitors, we offer bespoke software development, allowing you to tailor the solution to meet your specific needs - at a cost much cheaper than you might expect!

To request a quotation or find out more, simply contact us at the address below, or visit the Sentry-go web site at <http://www.Sentry-go.com/build-a-custom-monitor.htm> and tell us more about your requirement.

About 3Ds (UK) Limited

Established in 1998, 3Ds (UK) Limited is a software house & consultancy specialising in easy to use, affordable monitoring solutions for both Windows Desktops & Servers through our Sentry-go® technology.

As software consultants, we also specialise in the latest client/server, n-tier, web-based & .NET technologies. We work either independently or alongside your own developers, offering a number of key services vital to the successful **D**esign, **D**evelopment & **D**elivery of **S**olutions. We have many years of experience in implementing robust, flexible and efficient solutions into new environments as well as integrating the same solutions with client's existing & legacy systems.

Why Choose 3Ds ?

3Ds is proud to be an independent consulting & software house based to the south of London in the United Kingdom. We believe that this allows us not only to respond quickly and effectively to our customer's specific needs but also guarantees that you can reach the right person with the right skills at the right time. We won't pass you from team to team & person to person in order to get an answer!

Our Sentry-go monitoring solutions are used by companies world-wide. In addition to these, we also offer other a number of others services ...

Within the UK

- Software development using the latest Windows, .NET & web-based technologies.
- Consulting services for development, project, or support-related issues.
- Additional product training.
- Customised training - for Sentry-go software, O/S, Technical Support, programming & technologies etc.

World-wide

- Software customisations & enhancements, in order to integrate Sentry-go with your existing & in-house systems.
- Custom monitor development.

3Ds (UK) Limited is an
itSMF member.

itSMF
The IT Service Management Forum



3Ds (UK) Limited
Design, Develop, Deliver Solutions!

69, Esher Road,
East Molesey,
Surrey.
KT8 0AQ

<http://www.3Ds.co.uk>