CaoTester

User's guide

Version 1.0

January 10, 2006

[Remarks]

[Revision history]

Date	Version	Content
2006-01-10	1.0	First edition.

[Hardware]

Model	Version	Notes

Contents

1. Introduction

CaoTester is a CAO interface-mounted integrated test tool to execute methods of the provider.

1.1.1. Window layout

1.1.1.1. Main window

The image below shows the main window of CaoTester.

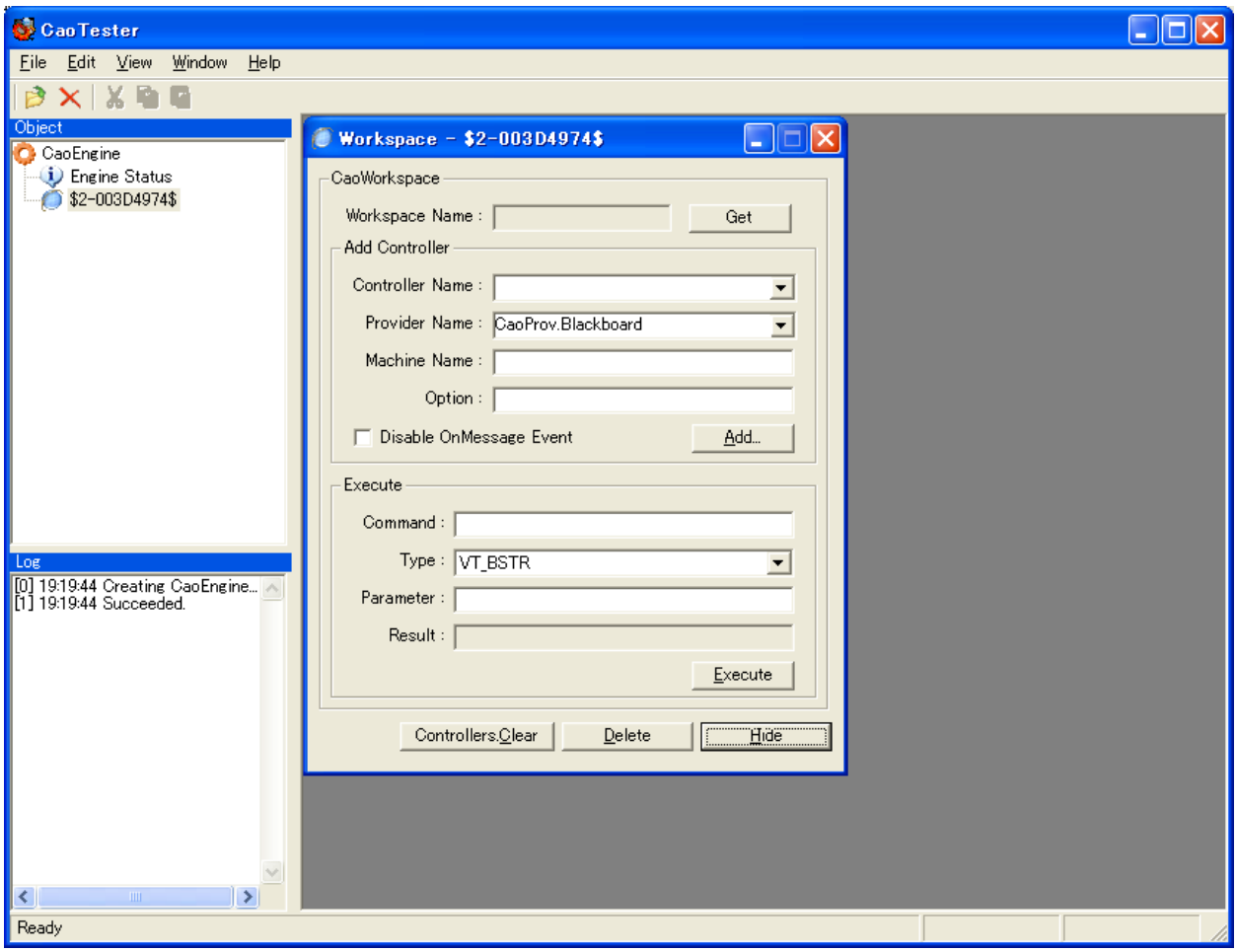

Figure1-1 CaoTester main window

The main window is composed of an object tree on the left, execution log under the left, and child window space on the right.

In CaoTester, each object has one child window, and these objects are managed by the tree view on the left of the window.

The result of a method executed by a child window and properties are displayed in the log window[1].

To output the log to a file, two ways are available: On the [File] menu, point to [Export], and then

[1] The execution result here indicates information such as HRESULT of the return value.

click [Log...]. Or, on the log window, right-click and point to [Export], and then click [Log].

1.1.1.2. Child window

The image below shows the Workspace window as an example of the child window.

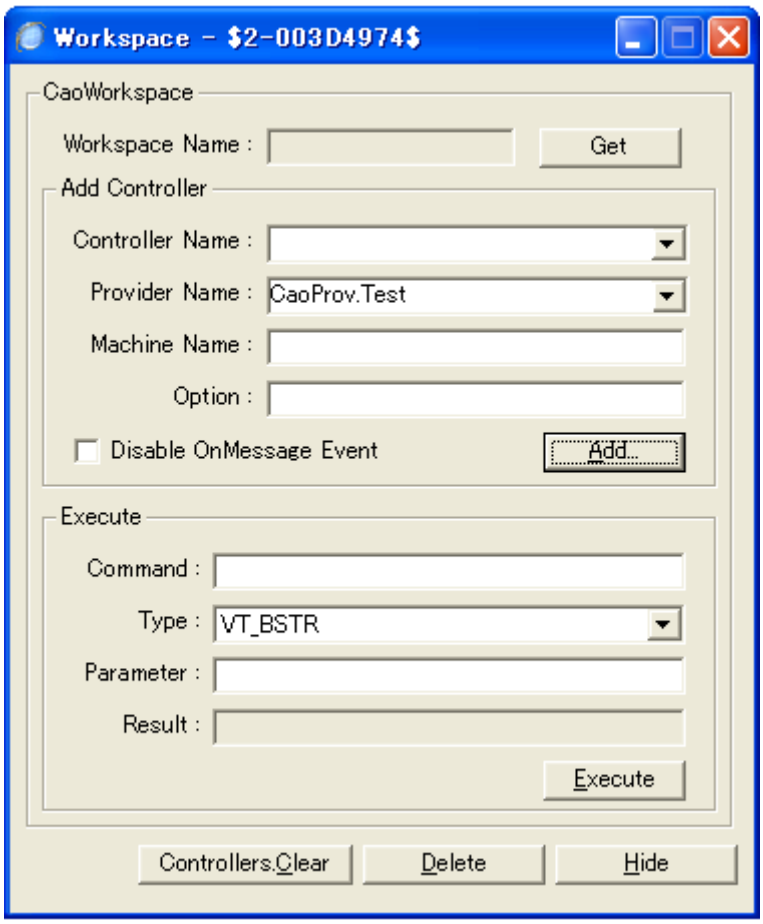

Figure1-2 Workspace window

A child window of CaoTester is created whenever an Add-type method of CAO succeeds. Most of child window has "Delete" and "Hide" button[2] on the bottom. "Delete" button discards the object, and deletes the window. "Hide" button deletes the window but the object will remain as is.

Each child window equips methods and properties of each CAO object[3]. Arguments for each method and properties are expressed as a textbox or a list box on the window. A text box or list box which background color is white represents input data, whereas with the gray color represents output data. Each method and properties is executed by pressing the button.

[2] Child windows of CaoEngine and EngineStatus do not have a "Delete" button.
[3] Because CAO collection-related methods and properties are wrapped, it is impossible to use them from CaoTester except for the Clear method.

Enter following items in Workspace window, and then click "Add".

Controller Name	:"Test"
Provider Name	:"CaoProv.DataStore"
Machine Name	: none
Option	: none
Disable OnMessage Event	: leave unchecked

Once "Add" is clicked, the method shown below is executed, and then a child window as Figure1-3 will be created. You can execute commands such as "AddVariable" or "Execute" in this window just like the Workspace window.

```
AddController(
        "Test"                  // <Controller Names :>
        "CaoProv.DataStore"     // <Provider Names :>
        ""                      //< Machine Names:> null character string
        "@EventEnable=true"     // <Option :>
)
```

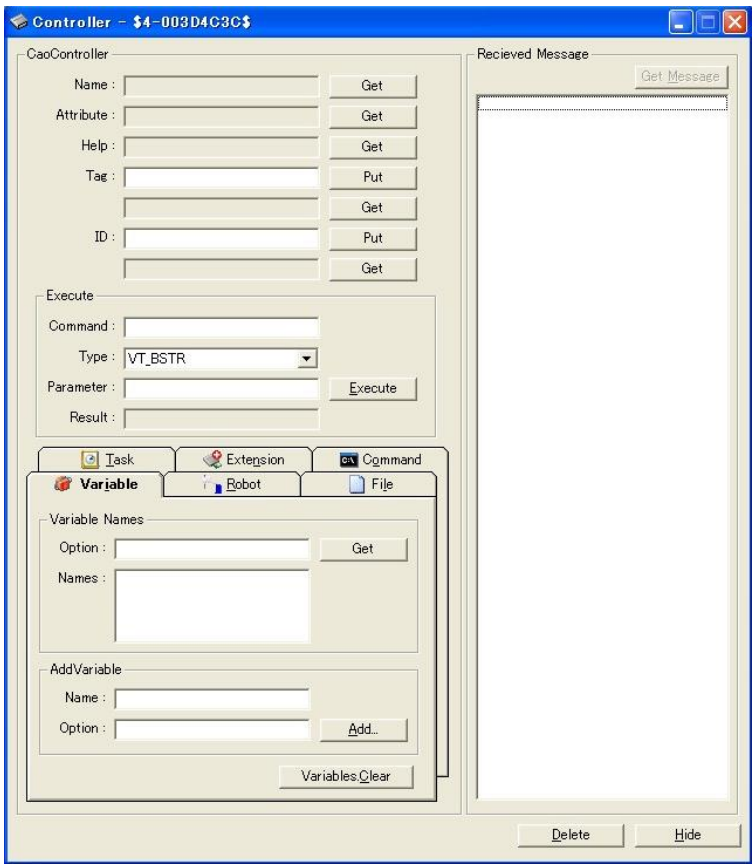

Figure1-3 Example of executing AddController

For an input where you can specify the data type, specify the value with RAC ddl (data description language). For details, refer to "Recording mode of 3.2 data" of 'RAC user's guide'.

The input example for "Figure1-2 Execute method "is shown below.

Ex.1)	Type: I2	Parameter: 100	Value: 100
Ex.2)	Type: BSTR	Parameter: Sample	Value: "Sample"
Ex.3)	Type: VARIANT	Parameter: (8, Sample)	Value: "Sample"
Ex.4)	Type: ARRAY|I2	Parameter: 100,200,300	Value: 100, 200, 300
Ex.5)	Type: ARRAY|VARIANT	Parameter: (8, Sample), (2,100)	Value: "Sample", 100

1.1.2. Special settings and functions

1.1.2.1. Event function ON/OFF

To set the value of "@EventEnable" option of AddController, use "Disable OnMessage Event" checkbox in the "AddController" area on the Workspace window.

Checking this check box will add "@EventEnable=False" to the Option argument of AddController, whereas unchecking this check box will add "@EventEnabled=True" to the Option argument of AddController.

Figure1-4 Disable OnMessage Event screen

1.1.2.2. EngineStatus window

The image below shows the Engine Status window.

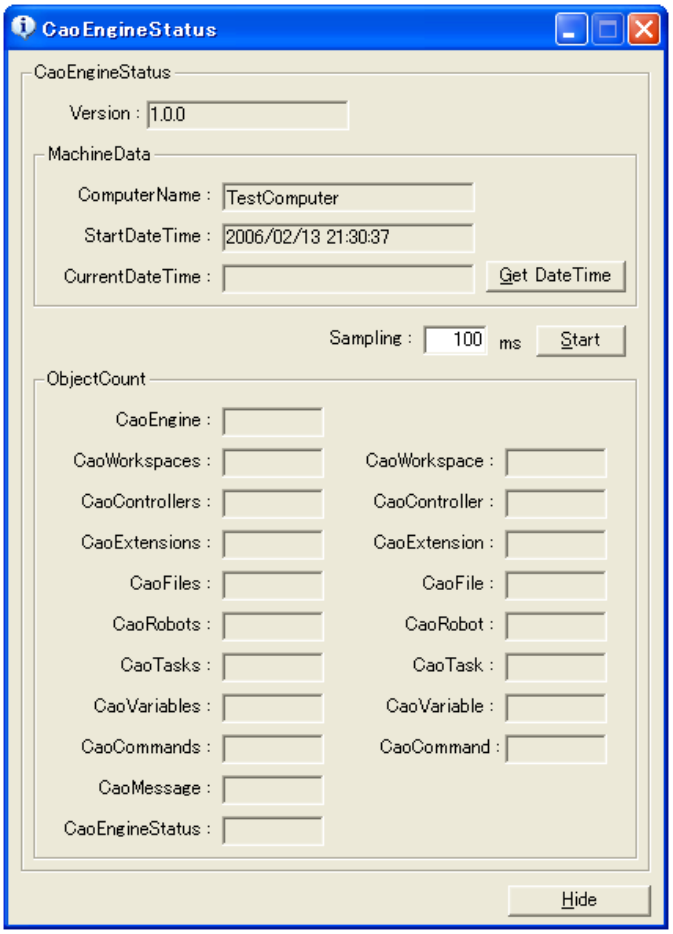

Figure1-5 EngineStatus window

When an window is generated, "get_Version", "get_ComputerName" and "get_StartDataTime" properties of EngineStatus are executed.

By clicking the "Start" button, "get_ObjectCout" property executes repeatedly at intervals of "Sampling" milliseconds.

1.1.2.3. Performance Check

This window enables you to check I/O performance of CaoVariable under the specified condition.

To display the Performance Check window, click the [Performance] button on the CaoVariable window.

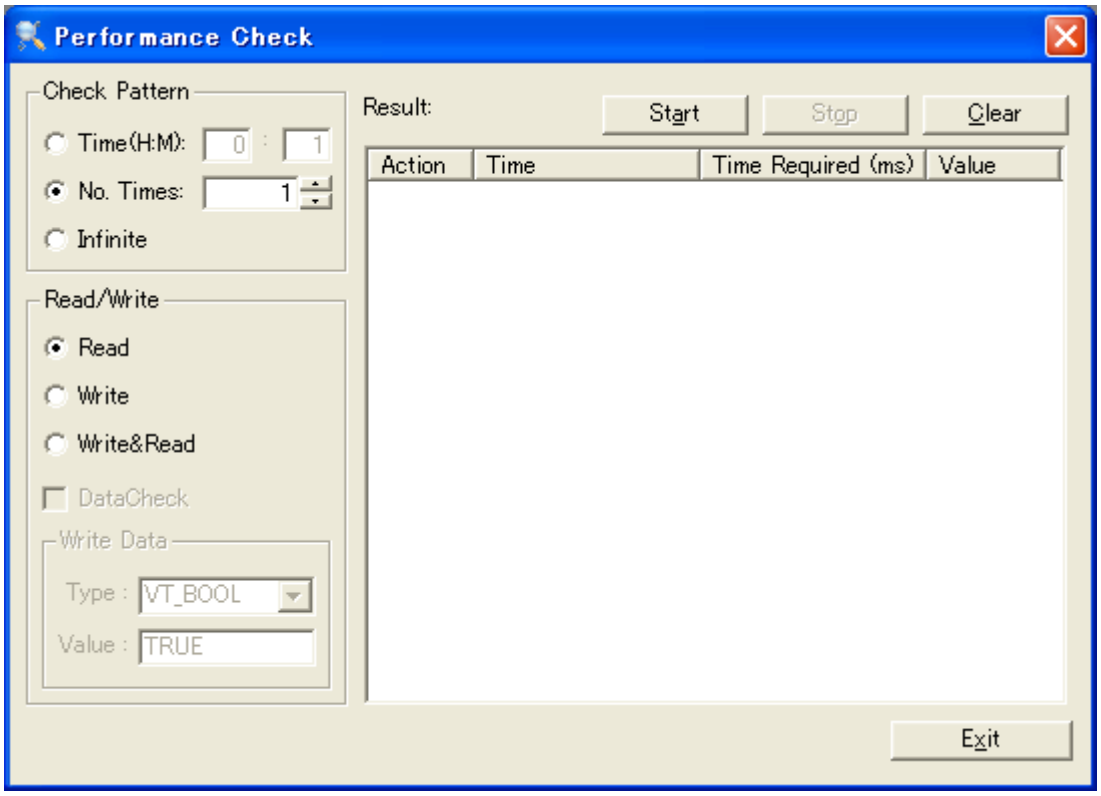

Figure1-6 Performance Check window

CheckPattern: Set the following items

- **Time (H: M):** Specify time period to operate the check.
- **No. Times:** Specify the frequency of the check.
- **Infinite:** When this option is selected, performance check keeps working until "Stop" is clicked.

Read/Write: Set the Reading/Writing operation

- **Read:** Read the specified data
- **Write:** Write the specified data
- **Write&Read:** Read and write the specified data.
- **DataCheck:** When this checkbox is selected, the data consistency check between the data reading and data writing are enabled. This option is available when "Write&Read" checkbox is selected.
- **WriteData:** Specify the type and value of writing data when "Write" or "Write&Read" is selected.

Result: Display the result of the performance execution in the grid. For about the each line name, refer to Table1-1.

Table1-1 Result row name list

Row name	Description
Action	Action is displayed. (START/END)
Time	Measurement start/end time is displayed.
Time Required(ms)	Measurement operating time is displayed.
Value	The written data is displayed.

Start: Start the measurement

Stop: End the measurement

Clear: Result grid is cleared.

1.1.2.4. CaoScript file export function

Operation of CaoTester can be saved as CaoScript file.

There are two ways to save the operation: On the menu bar, point **[Edit]**, and then click **[Start Recording]**, **[Pause Recording]** or **[Stop Recording]**. Or, on the tool bar, click CaoScript recording control button (see Figure 1-8).

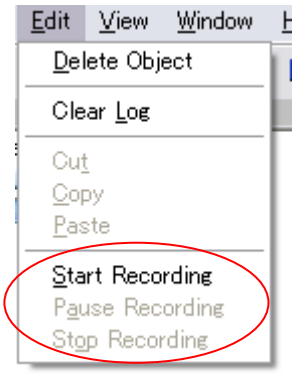

Figure1-7 CaoScript recording control menu

Figure1-8 CaoScript recording control button

Export the recorded operation to the CaoScript file. From the menu bar, point to **[File]**, and then click **[Export Script...]**.

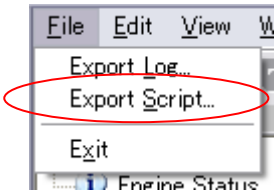

Figure1-9 CaoScript file export menu

Note that the following restrictions are applied.

* You must specify the controller name at time of AddController execution.

* Following methods and properties are not exported.

- Engine object: All methods
- Workspace object: All methods and properties except for AddController
- EngineStatus object: All methods/All properties[4]
- OnMessage event
- Message object: All methods/All properties[5]

[4] This is because CAO object, which is CaoScript embedded object, is used.

[5] This is because Message object is not supported by CaoScript.