

## **Run Menu**

### **Stop**

Stops the execution of an IconAuthor application; but does not exit the application.

### **Go**

Begins or resumes the execution of an IconAuthor application.

### **Step**

Executes the next icon and then stops.

### **Jump**

If the next icon is a single icon, executes the next icon and then stops. If the next icon is a composite, a SubApp, or a Subroutine icon, execution flows through the entire composite, SubApp, or a Subroutine and then stops.

### **Halt Application**

Cancels the execution of an IconAuthor application and returns focus to the IconAuthor.

### **Exit**

Exits IAScope.

## **View Menu**

### **Ribbon**

When this option is toggled on, the ribbon bar is visible.

### **Application Status**

When this option is toggled on, the status of the icon currently being executed is visible.

### **Data**

When this option is toggled on, the name and content of the icon being executed is visible.

### **Translated Data**

When this option is toggled on, the name and content of the icon being executed is visible. When an icon uses a variable, instead of a fixed value, the translated data bar shows the value currently stored in the variable instead of the variable name.

### **Memory**

When toggled on, the current amount of memory is visible.

### **Status**

When this option is toggled on, the status bar is visible.

## **Options Menu**

### **Always On Top**

When this option is toggled on the IAScope window resides on top of all other windows.

### **Live Updates**

When this option is toggled on the application status is updated as each icon is executed.

### **Beep On Stop**

When this option is toggled on a beep is sounded every time the IconAuthor is stopped by IAScope.

### **Save Settings On Exit**

When this option is toggled on and you exit from IAScope, settings that describe the condition of the program (such as the always on top, and beep on stop) are remembered the next time you restart the program.

## **Help Menu**

### **Index**

Displays an alphabetical list of all Help topics that are available.

### **Keyboard**

Displays the accelerators for performing some tasks with the keyboard instead of the mouse.

### **Commands**

Displays an explanation of commands.

### **Procedures**

Displays a description of how to use IAScope

### **Using Help**

Displays a short tutorial and other information about using Windows online Help.

### **About SmartObject...**

Displays IAScope copyright and version information.

## The IAScope Window

The following IAScope window contains the following components:

Title Bar

Menu Bar

Ribbon Bar

Application Status Bar

Data Bar

Translated Data Bar

Memory Bar

Status Bar

**Title bar**

The title bar contains the application name "IAScope."

## **Menu bar**

The menu bar contains the four IAScope menus. The Run menu lets you control how an IconAuthor application executes. For example, you can execute the application as it would normally execute or you can step through the execution process one icon at a time. The View menu lets you customize which features of IAScope are visible. The Options menu lets you control certain characteristics of IAScope such as whether it appears on top of other windows. The Help menu lets you access online information on how to use IAScope.

**Ribbon bar**

The ribbon bar contains buttons that allow you to perform the most frequently used commands that are available through the menu bar.



## **Application status bar**

During execution, the application status bar provides information about the icon currently being executed. For example, it tells you which icon is being executed and the icon's position within the structure.

**Data bar**

The data bar contains name and the content of the icon currently being executed.

**Translated data bar**

The translated data bar is similar to the data bar. It too provides information about the content of the icon being executed. However, if an icon uses a variable instead of a fixed value, the translated data bar shows the value currently stored in the variable instead of the name of the variable.

**Memory bar**

The memory bar shows the current amount of free memory and percentage of free resources.

**Status bar**

The status bar displays a brief description of the menu commands and ribbon bar buttons. For a description of a menu command highlight the command in which you are interested. For a description of any active button in the ribbon bar position the cursor over the button.

**Note:** Most of the bars that make up the IAScope window can be turned on or off via options in the View menu.

## Using IAScope to Control Execution

Before you use IAScope run IconAuthor and open the IconAuthor application you are interested in debugging. When you run IAScope you'll find that by default, it always stays on top of other programs. This is so you can see what's going on in the IAScope window at all times. (If you need to turn off the Always On Top feature, choose Always On Top from the Options menu to de-select it.)

Make sure IAScope is on top of the IconAuthor application you want to view before proceeding.

When you are ready to debug your IconAuthor application you begin by running it in the conventional way. That is, you click on one of the Run buttons in the IconAuthor window ribbon bar.

Notice that even though you clicked on a Run button the application does not immediately run when IAScope is running. This is because the IAScope Stop button is activated by default. Once the IconAuthor Run button is selected, IAScope takes control of the application's execution. Although the application is not yet executing, the information in the IAScope window has changed to show that

Initially, the application status bar indicates that execution has "Stopped" at the Start icon which is located in row 1, column 1. At this point, you have the ability to use the Go, Step, Jump, and Halt buttons to control how the application executes.

You can also use the Debug menu (in the IconAuthor Run menu) to set and clear stop points in execution.

## **Go Button**

The Go button starts standard execution. Standard execution means that the application executes just as it does in IconAuthor and Present; icons automatically execute one after another unless an Input or InputMenu icon in the application is awaiting information from the user. The information in the IAScope window changes fairly rapidly when you choose the Go button.

Once you have clicked on the Go button you can stop execution at any point by clicking on the Stop button. Execution stops at the current icon and information about the icon is displayed in the IAScope window. To resume execution use the Go, Step or Jump buttons.

To escape from the running IconAuthor application, press the Escape key or click on the Halt button.

## **Step Button**

The Step button executes the next icon and stops.



## **Jump Button**

The Jump button has features similar to both the Go button and the Step button. It executes the next icon and then stops or proceeds automatically depending on the particular icon being executed. If the next icon is a single icon, execution stops (just like the Step button). If the next icon is a composite, a SubApp, or a Subroutine icon, execution flows through the entire composite, sub-application or subroutine as it does during standard execution.

## **Halt Button**

Clicking the Halt button when an IconAuthor application is running has the same effect as pressing the ESC key; the application stops running.

## Using Information in IAScope

There are several types of information available through the IAScope window. You use the IAScope buttons to control how execution proceeds. Whenever execution is stopped the IAScope window contains information on the next icon to be executed.

By default, during execution this kind of information is displayed for each icon as rapidly as the icon is executed. If you prefer not to have this kind of information flash by, choose Live Updates from the Options menu to de-select the option. When the Live Updates option is not in effect information on an icon only appears when execution stops.

**Hint:** IAScope doesn't have to display information you're not interested in. You can use the View menu to turn most areas of the IAScope window on or off as necessary. For example, to turn off the memory bar choose Memory from the View menu.

Related Topics:

[Viewing Variables](#)

[Viewing the Stack](#)

## Viewing Variables

If your application uses variables you can use the IAScope window to check the value stored in a variable at any point during execution.

The data bar shows when an icon uses a variable, such as @GRAPHIC, instead of a specific filename. The translated data bar just below, shows the value that is currently stored in that variable.

Related Topic:

[Setting and Clearing Stop Points](#)

## Viewing the Stack

Many authors create applications that are modular in that they are made up of multiple sub-applications and subroutines. As necessary, these files can call one another in the course of the application. Behind the scenes, IconAuthor tracks the execution history (called the **stack**) of SubApp and Subroutine icons, as well as the loops that occur, so that IconAuthor can "return" either from calls to a file or from a loop. IconAuthor can handle up to 99 subroutines, sub-applications and loops that are executed and not returned from. If your application exceeds this number the system runs out of memory and a STACK OVERFLOW message occurs.

As you run your application through IAScope use the information in the Stack field to help avoid STACK OVERFLOW messages. Use IAScope to monitor the number of stack entries at different points during execution. If you detect that a problem may be occurring, one solution is available through the SubApp icon restart command. A SubApp icon can specify a restart command to remove all previous calls to sub-applications and subroutines and all previous loops from the stack.

## Setting and Clearing Stop Points

Use the commands in the IconAuthor's Run (Debug) menu to set and clear specific stop points. Set a stop point by clicking on an icon in the structure and choosing Set Stop Point from the Run (Debug) menu. A small stop sign symbol appears next to the icon to denote that it is a stop point.

The stop point means that when you run the application, it will stop execution at this point. You can view the current information in IAScope and then click on IAScope's Go, Step, or Jump button to resume execution.

To clear a stop point, click on the icon and choose Clear Stop Point from the Run (Debug) menu. Similarly, use the Clear All Stop Points command to remove all stops.

## **IAScope Contents**

The Contents lists the Help topics available for IAScope. Use the scroll bar to see entries not currently visible.

### **Keyboard**

[IAScope Keys](#)

### **Commands**

[Help Menu](#)

[Options Menu](#)

[Run Menu](#)

[View Menu](#)

### **Procedures**

[Setting and Clearing Stop Points](#)

[The IAScope Window](#)

[Using IAScope to Control Execution](#)

[Using Information in IAScope](#)

[Viewing the Stack](#)

[Viewing Variables](#)

## IAScope Keys

The following accelerator keys and key combinations are available in IAScope:

<b>Menu</b>	<b>Command</b>	<b>Key(s)</b>
Run	Stop	Ctrl + P
	Go	Ctrl + G
	Step	Ctrl + S
	Jump	Ctrl + J
	Halt Application	Ctrl + H

## IAScope Commands

To get help with a command, choose the appropriate menu.

### Run Menu

- Stop
- Go
- Step
- Jump
- Halt Execution
- Exit

### View Menu

- Ribbon
- Application Status
- Data
- Translated Data
- Memory
- Status

### Options Menu

- Always On Top
- Live Updates
- Beep On Stop
- Save Settings On Exit

### Help Menu

- Index
- Keyboard
- Commands
- Procedures
- Using Help
- About IAScope...



## **IAScope Procedures**

[Setting and Clearing Stop Points](#)

[The IAScope Window](#)

[Using IAScope to Control Execution](#)

[Using Information in IAScope](#)

[Viewing the Stack](#)

[Viewing Variables](#)

