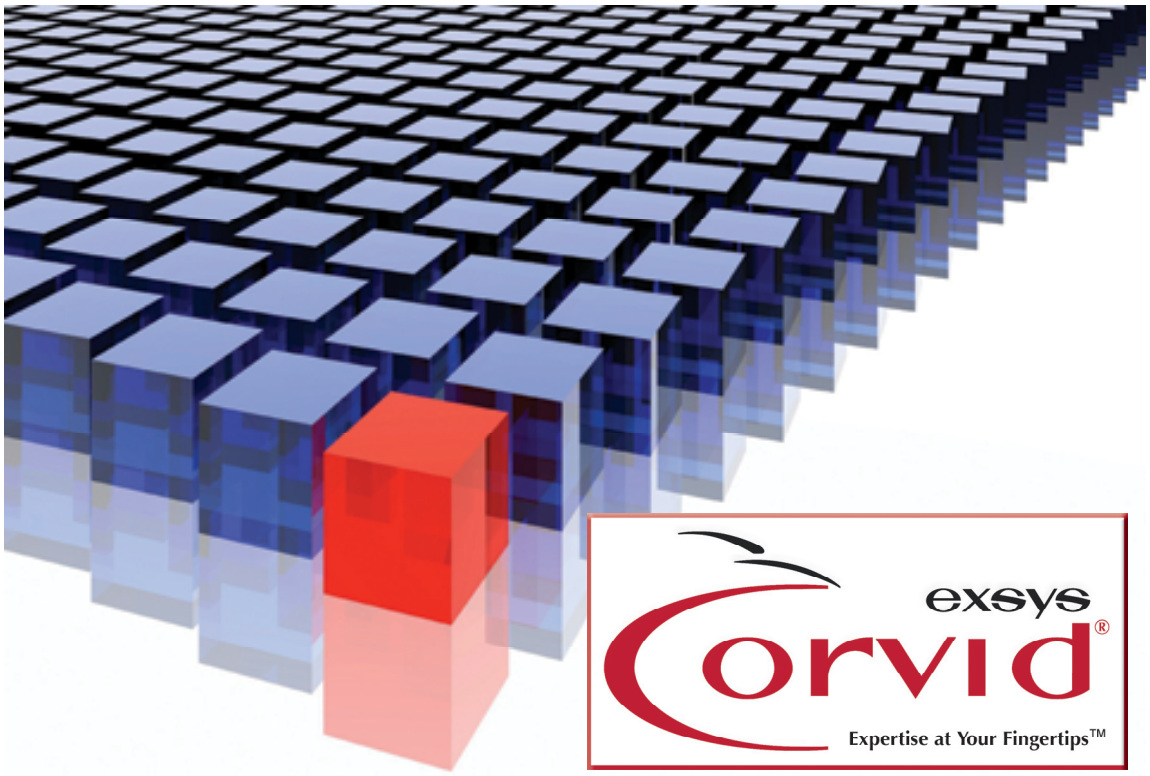


Exsys Corvid v.4 New Features

Exsys Corvid v4 is enhanced in many ways to make the program both more powerful and even easier to use.

- The new Action Block feature provides a new way to build systems, and is especially well suited to “Smart Questionnaires”.
- A validation capability makes it easy to run large numbers of tests on a system and to look for various types of problems.
- Spell checking catches typos as text is entered
- New “How To” documents explain the details of implementing advanced features in your systems
- A new manual is easier to work with and directly tied to the “Help” button on many windows

Action Blocks

Action Blocks provide a new way to describe decision-making logic. They are an alternative to Logic Blocks for certain types of systems, and can often be used in conjunction with Logic Blocks.

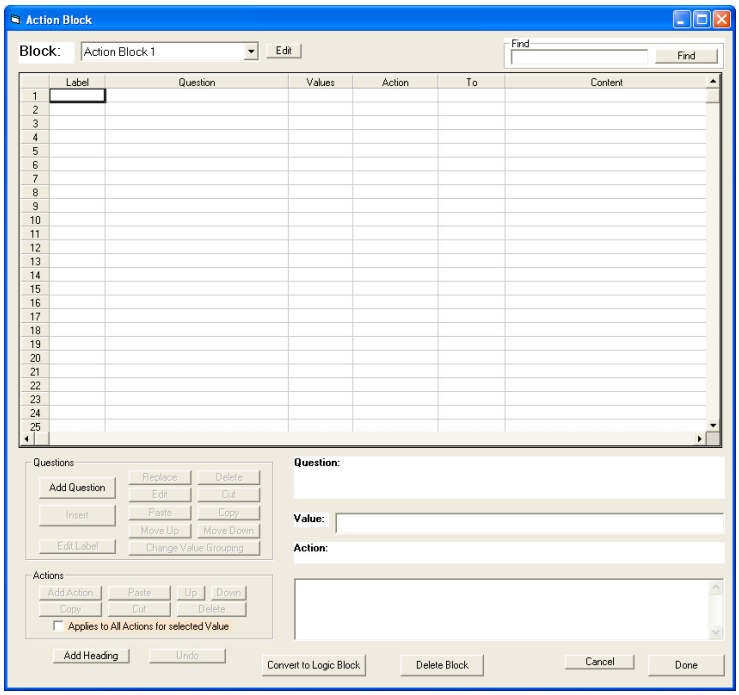
Action Blocks are fully described in Chapter 8 of the new Corvid Manual, along with an Action Block tutorial.

Action Blocks are a way to represent simple forward-chaining logic. They use a spreadsheet style approach that is well suited to many procedural systems. They are ideal for smart questionnaires, surveys, dichotomous keys and in systems where the decision-making logic is not deeply nested.

Action Blocks contain a series of questions, along with either possible values that may be selected, or Boolean expressions that will be true or false based on the user's input. Each value/expression can have one or more associated actions to take, such as setting values, skipping over some questions, running other Logic, Action or Command Blocks, executing Corvid commands, etc. This is a very simple way to describe logic, and it is applicable to a wide range of problems – especially when combined with the capability of Logic Blocks where needed.

To add an Action Block to a system, click the "A" block icon.

This will open a new Action Block:

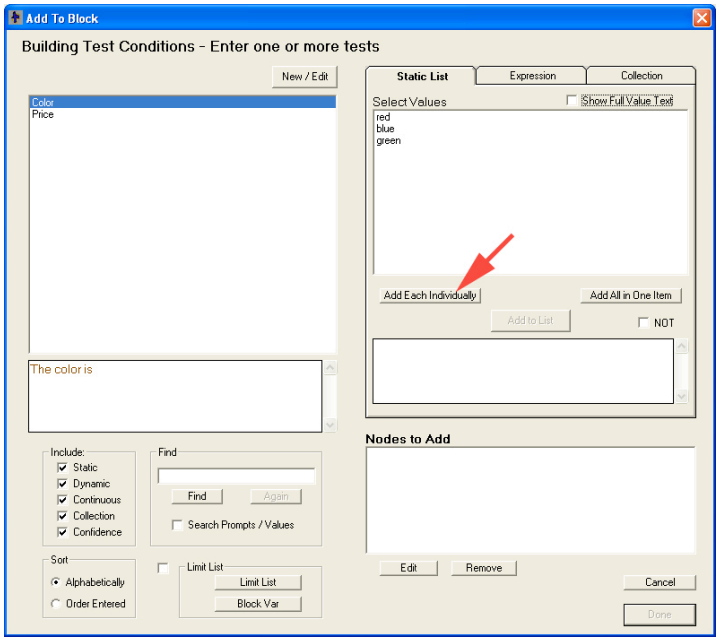


An Action Block has 6 columns:

Label	A unique label for the question. This is used in "GoTo" actions that skip over some questions.
Question	The question to ask of the end user. This is the prompt of the variable being used.
Values	For Static List variables, the possible values for the variable. These may be grouped if several values should produce the same actions. For other types of variables, this column will list one or more Boolean expression built using the variable. Based on the user input, these expressions will be true or false. The expressions do not have to be mutually exclusive (more than one can be true at the same time).

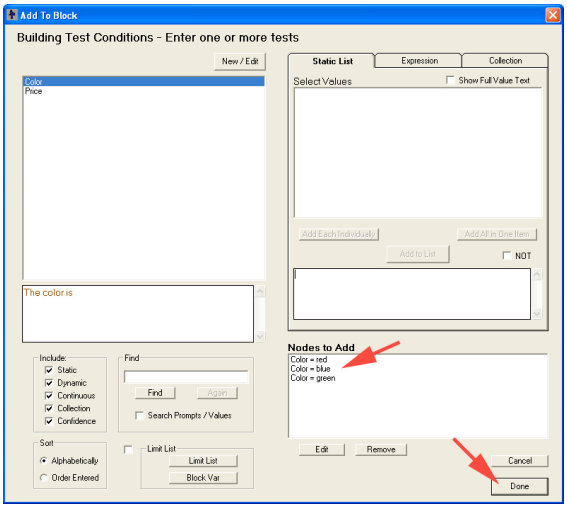
Action	The action to take when the value/expression in the Value column is true. A single value may have multiple associated Actions. There are various types of actions, which may require one or two additional items of data.
To	If the Action is done to something, it is listed here. For example, an action to execute a Logic Block, would list the Logic Block to run in the To column.
Content	Some actions require both a To and a Content value. For example, if the action is to set the value of a variable, the To column would be the variable and the Content column would be the value to assign.

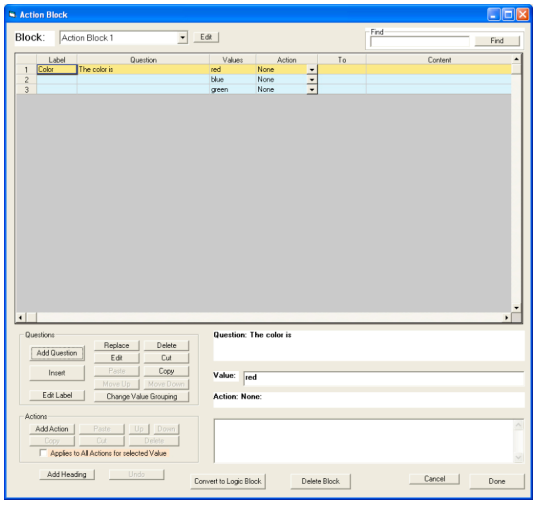
To add the first question to the block, click the “Add Question” button. This will display the same window for adding conditions as is used in Logic Blocks.



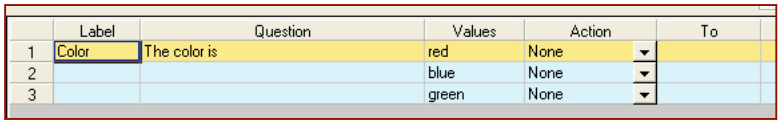
The various Values are selected, just as when adding nodes to a Logic Block. For example, if the “Add Each Individually” button is clicked for a Static List variable, each value will be added individually.

Click the “Done” button to add the list to the Action Block.



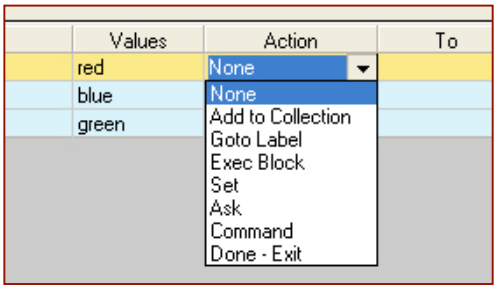


Looking at the color Question:



	Label	Question	Values	Action	To
1	Color	The color is	red	None	
2			blue	None	
3			green	None	

To select an Action to take if the Value is selected, click the drop down arrow in the associated Action cell:



This will display the possible actions that can be taken:

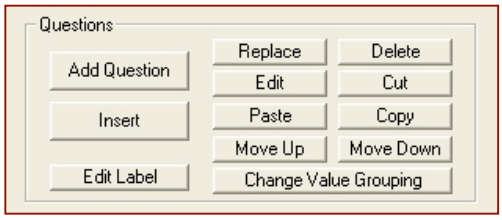
None	No action will be taken if the value is selected.
Add to Collection	Add a string to a collection variable. This is a convenient way to build up text based on the user's input. This text can be simple statements for a report, or complex HTML commands to dynamically build a custom web page for the user. The Collection variable to use is entered in the To column and the text to add is entered in the Content columns.
GoTo Label	Skip to the specified label. This allows skipping over questions. If the user's input indicates that some of the following questions are not needed, this is a convenient way to jump over them. You can only use GoTo to move down the list – not back to a previous question. The label to go to is selected from a drop down list in the To column. Since the next question in the list is automatically the next one to ask, the Goto label list only includes questions more than one question down in the list.
Exec Block	Immediately execute the specified Logic, Action or Command Block. The block to execute is selected from a drop down list in the To column.
Set	Assign a value to a variable. This can be used to set the value to any variable in the system. If a variable has a value set, and the variable appears later in the list, the assigned value will be used rather than asking the user for input. The variable to assign to is entered in the To column and the value to assign is entered in the Content column.
Ask	Ask the end user for the value of a variable. This can be used to obtain additional information that is not explicitly used as part of the system logic, but which may be included in reports etc. The variable to ask is entered in the To column.
Command	Execute a single Corvid command. The command is built with the Command Builder window and entered in the Content column.
Done - Exit	Exit the Action Block even if there are more questions in the block

When an Action is selected, the To and Content columns will be highlighted in red to indicate that they require input and, in many cases, will change to drop down lists of the valid selections.

Question Buttons

When a cell is clicked in the block, it will select a question and an action. Highlighting colors will indicate the question, value and action(s) selected. (These highlight colors can be changes using in the "General" tab of the Properties window.)

There is a group of buttons for adding/editing questions in the block.



Add Question	Add a new question to the end of the list as the last one.
Insert	Add a new question at the currently selected row in the block.
Replace	Add a new question replacing the currently selected question.
Delete	Delete the currently selected question.
Edit	Open the Variable window to edit the Prompt and other properties of the variable associated with the selected question.
Cut	Delete the currently selected question, but keep a copy that can be pasted.
Copy	Make a copy of the selected question that can be pasted.
Paste	Paste the question from the previous Cut or Copy.
Move Up	Move the question and all values up in the list.
Move Down	Move the question and all values down in the list.
Edit Label	Edit the selected label. This is active only if the click to select the question was on the Label column.
Change Value Grouping	This allows reordering, combining or splitting up groups of values.

Action Buttons

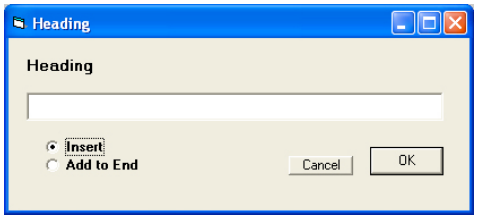
There is also a group of buttons for adding and editing Actions:

Add Action	Add a “None” action below the currently selected action.
Copy	Copy the current action to be pasted.
Cut	Delete the current actions, but keep a copy that can be pasted.
Paste	Paste the Action from the last Cut or Copy.
Delete	Delete the selected Action.
Up	Move the Action up in the list. This applies only when a Value has more than one Action.
Down	Move the Action down in the list. This applies only when a Value has more than one Action.
Applies to all Actions for Selected Value	This checkbox makes the Cut, Copy and Delete buttons apply to all the Actions for the selected Value.

Add Heading Buttons

The Add Heading button will add a heading line. Heading lines have no effect on the logic, but makes it easier to segment the questions. Clicking the button will display the heading window.

Enter the heading to add, and select to add it to the end or to insert it at the currently selected question.



Undo Button

The Undo button allows you to undo your last actions. You can undo up to 5 steps back.

Convert to Logic Block Button

The Convert to Logic Block button will permanently convert the Action Block to an equivalent Logic Block. This will allow you to edit and enhance it as a Logic Block. Once converted, the Logic Block cannot be converted back to an Action Block.

Delete Block Button

The Delete Block button will delete the current Action Block.

Validation Testing

Any Corvid system should be tested thoroughly before fielding it. The new Validation feature is designed to simplify this. Corvid will automatically run a system many times, systematically setting values for variables and checking for various types of problems. Specific tests can be specified and Corvid will check each run for these. For example, if a system should always have at least one Confidence variable with a value greater than 0, the validation tests can include checking for any cases where this is not true. Validation is a very flexible way to systematically test all or part of a system. Validation test parameters can also be saved to a file and rerun in the future to recheck a system, or to compare output files to make sure a system is producing the same output.

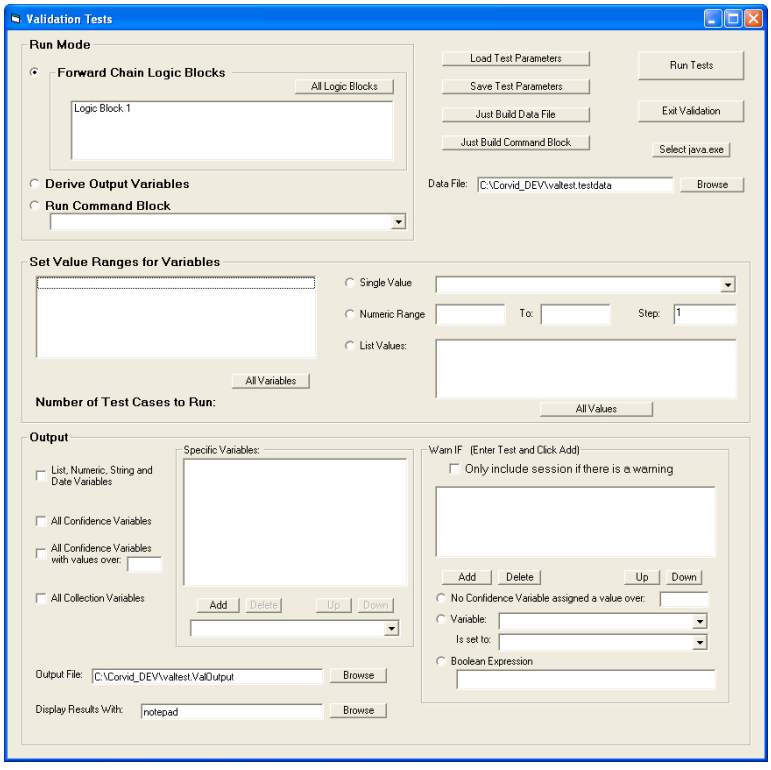
The detailed instructions for running validation tests are in Chapter 13 of the new Corvid manual. That chapter should be read before using the Validation function.

To run validation tests, select "Validation" from the "Run" menu.

Validation tests can:

- Run specific logic blocks in forward chaining.
- Derive the values of specific variables
- Run a specific Command Block

Once the run mode is selected, Corvid will display the variables that are relevant to the IF conditions selected. Additional variables can be added if needed for very complex systems.



For each variable, a range of values is specified for the tests. For static list variables this can be one or more of the values. When there are multiple values, they will be tested one at a time. For numeric variables, there can be a range of values from stating value to ending value with an optional increment value (an increment of 1 is used if none is specified.) Numerics can also have a single value specified. For all other types of variables, a single value is used.

Corvid will systematically test all of the possible combinations of values in the ranges selected. As the ranges are set for each variable, Corvid will display the total number of tests in the run. If this becomes a very large number, Corvid will warn you. Depending on the size of the system and speed of computer, the validation tests typically run on the order of 1-10 every second. So, tens of thousands of tests may take hours, hundreds of thousands may take days. However, the actual time to run the tests is very dependent on system capabilities.

A file of results is produced by the various tests. The output will always include the input values set. In addition it can include the values set for:

- All List, Numeric, String and Date variables
- The value for all Confidence Variables
- The value for all Confidence Variables over a threshold value
- The value for all Collection Variables
- The value for specific variables that have been added to the “Specific Variables” list

This produces a file with all of the input and output of the many validation tests. This file can be saved, along with the validation parameters. At a later time, the same validation tests can be run and the output files compared to see if there have been any changes in system output.

In addition to the input/output file, warnings tests can be set to check for various problems.

- No Confidence Variable has a value over x
- A variable has a specific value
- Any Boolean test

Enter the test and click the “Add” button to add it to the list. There can be as many warning tests as desired.

An “Only Include Session if there is a Warning” checkbox will cause the validation output file to include ONLY sets of data that produce a warning. This makes it easy to see if there are problems, and check the input that produced them.

Corvid Manual

Corvid v4 includes a new Corvid manual. This takes the earlier manual and combines in all the updates and addendums. It also adds new chapters on Action Blocks (with tutorial) and Validation. The manual has been reformatted to make it easier to use.

New Help System

Most windows in Corvid v4 have a “Help” button. This will open an Adobe Acrobat Reader window and display the section of the manual related to that window. This provides a more complete description of all the functions than the previous help system, and allows for rapid checking other sections of the manual, searching for terms, etc.

Corvid “How To” Documents

A set of “How To” documents has been created with step-by-step details on how to implement some of the more advanced Corvid capabilities. In most cases a sample system illustrating the techniques is provided, along with the code for the sample system. Where practical, the sample code can even be merged with your system to add the functionality.

"How To" documents cover topics such as automatically saving user input to a database, working with frames to display helper pages, using CSS and Javascript in Corvid Servlet Runtime templates and many other topics. A folder with the "How To" documents is created when Corvid is installed, but new ones are constantly being added. The current "How To" documents and sample systems can also be found at:

http://www.exsys.com/support/howto

Exsys Corvid How Tos	
Building Wink "What I Need to Know" Site Guides	Build a virtual concierge for complex web sites to dynamically build a custom page of content and links based on the visitor's interests.
Automatically Save User Input to Database	Build systems that automatically save all user input to a database, allowing the user to exit a run at any point and restart later where they left off. Explains how to use this approach for both applet and Servlet Runtimes.
Add a Button to save User Data and Exit Run	Allow the user to exit a session by clicking on a "Save" button that will save their input and exit the session. Used when saving data after every question is impractical. (For Applet and Servlet Runtimes)
Save User Input to a File when Running as an Application	Used to save user input when running standalone rather than online applications.
Techniques for Interfacing to Databases	Various ways to interface to a database to read and write data.
Options in Running a System During Development Testing	Various ways to test run a system and how to use Properties options.
Using Cascading Style Sheets (CSS) with Corvid Servlet Templates	Ways to apply CSS to control the look-and-feel of a system delivered with the Corvid Servlet Runtime.
Creating a Desktop Shortcut to run a Corvid System Standalone	Setting up the Java parameters to build a shortcut that will call Java to run a Corvid system as an application.
Designing Systems to Run in Kiosk Environments	Creating controls that can be run using a touch-screen, including emulating a keyboard in Corvid.
Displaying Instructional "Helper pages using Framesets – Applet Runtime	Display an HTML page of instructions, explanation or reference material that may be needed to answer a question.
Displaying Instructional "Helper pages using Framesets – Servlet Runtime	Display an HTML page of instructions, explanation or reference material that may be needed to answer a question.
Obtaining a Continuous Value from Image Maps	Using an image map to represent one or two-dimensional ranges of values. The spot where the ser clicks is returned as a value proportionately indicating where on the map they clicked. This allows some of the functionality of a "slider bar" to be added to a system.
Setting a Dynamic List Variable Value List from a Database	Initializing the values that can be selected from a Dynamic List by using a database call.
Using Javascript in Servlet Templates for Special Effects	Using Javascript to validate input, position the cursor and perform other actions.
Expanding the Applet Tag to Fill Window	Explains settings that will make the applet window fill the entire browser window. This allows a system running with the Applet Runtime to look much more like a Servlet Runtime system.

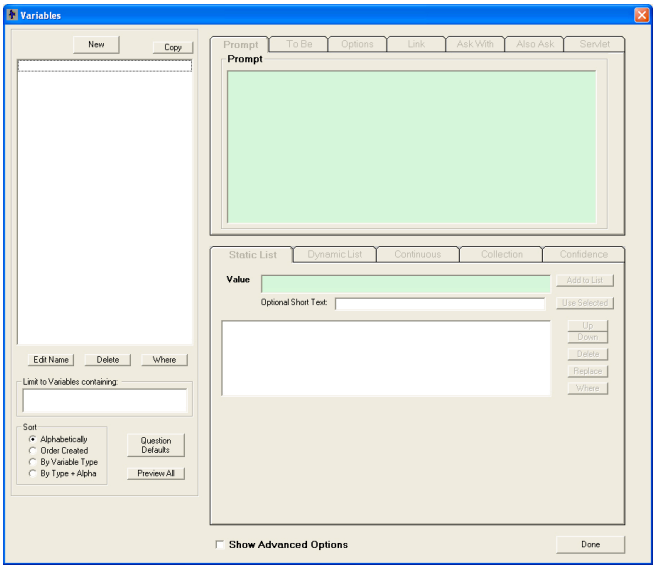
Adding Variables

The window for adding variables has been simplified.

In most cases when adding a new variable all that is needed is the Prompt text, and for Static List variables the associated values. These fields have been highlighted in light green. Just enter the Prompt text.

Depending on the type of the variable (Static List, Numeric, Date, etc.) the appropriate tab will be selected to set the option for that variable.

To see the full set of options, just click the "Show Advanced Options" checkbox at the bottom of the window.



Adding Static List Variables

For Static List variables, enter the first value text in the lower green box and click the "Add to List" button, or just click the "Enter" key. Multiple values can be input by entering text and clicking "Enter" for each value. Once a value is entered, the cursor automatically resets in the green value edit box. The values entered will appear in the list.

Each value can also be given an optional "Short text" that will be used in the blocks. This is convenient if the value text is long and would make the trees difficult to read. The short text is limited to standard text characters. The following characters cannot be used:

[~!@^&*()+="" '><./:;{}? |\`]

These illegal characters will be deleted from the short text if they are used. A space is illegal, but it will be converted to an underscore (_) character rather than deleted.

The short text is optional, and if not specified, the full value text will be used after any illegal characters are automatically deleted.

It is often convenient to use a section of the full value text as the short text. To do this:

- Enter the full value text.
- Use the cursor to highlight the portion to be used as the short text.
- Click the "Use Selected" button next to the Short Text edit box.

If a value has a short text that is other than the full text striped of illegal characters, the list of value will display:

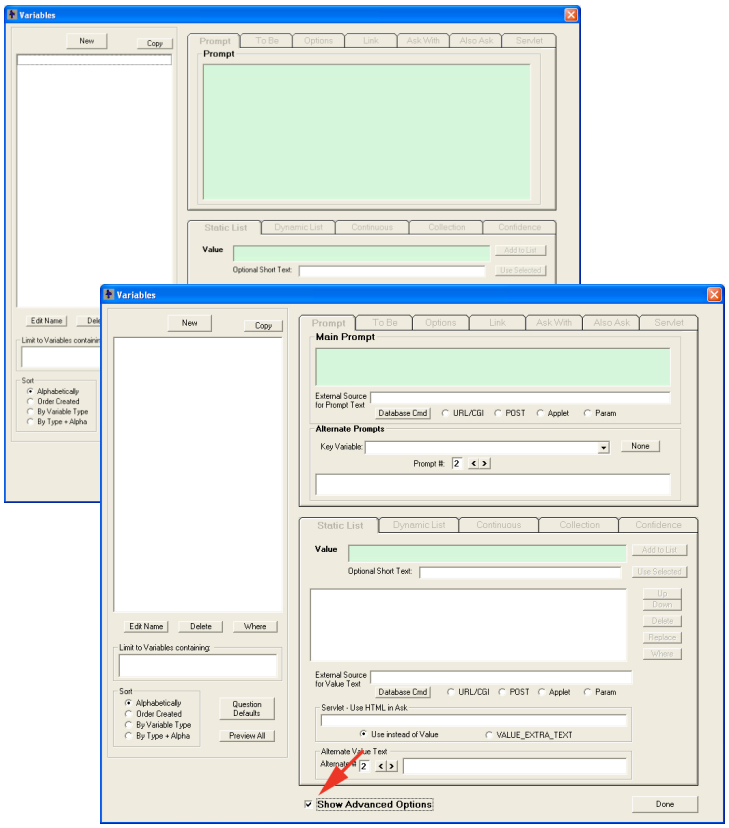
Short_text : Full text

If there is no special short text, the list will have only the full text.

Advanced Options

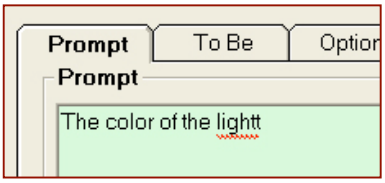
The many advanced options for prompts/values in multiple languages, or obtaining the text from a database have been hidden. These are rarely needed and new users are prone to unintentionally click on advanced controls. Likewise, many of the other advanced options on the other tabs are also hidden.

If you need the advanced options, simply click the “Show Advanced Options” button at the bottom of the window to see the full range of commands.



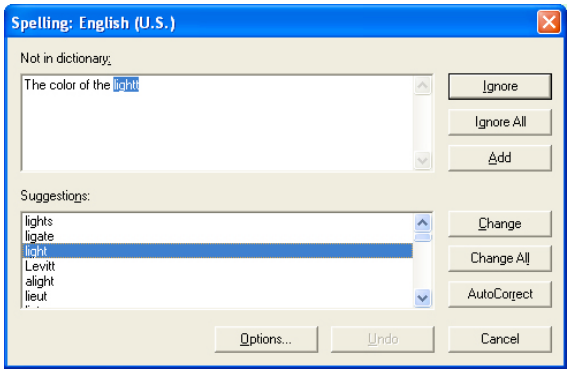
Spell Checking

Another new feature in Corvid v4 is a spell checking function for edit boxes where standard English text is expected. These include the Prompt and Values of a variable, text added to a collection variable and others.



As text is entered in these edit boxes, if any words appear to be incorrectly spelled, they will be highlighted with a wavy red line.

Misspelled words can be directly fixed in the edit box, or you can click **F7** to display the spell checker window.



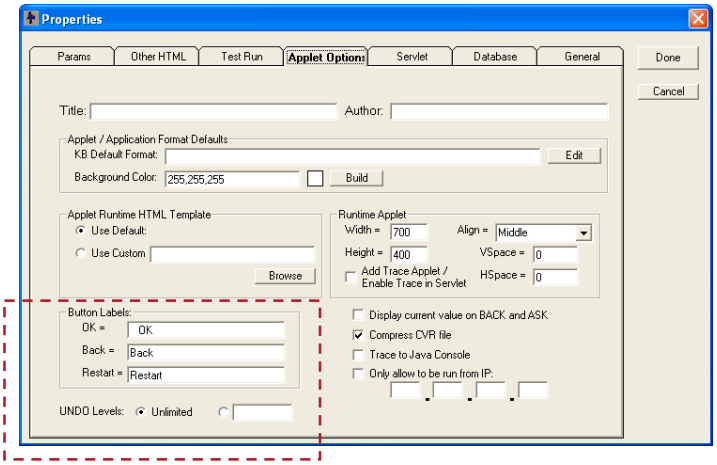
This window makes suggestions and allows editing, ignoring and seeing the misspelled words.

The spell checker dictionary is for English. If you are using another language or do not wish to use the spell checker, deselect the “Enable Spell Checking” checkbox in the “General” tab of the “Properties” window.

Other ver 4.0 Commands

Undo Levels

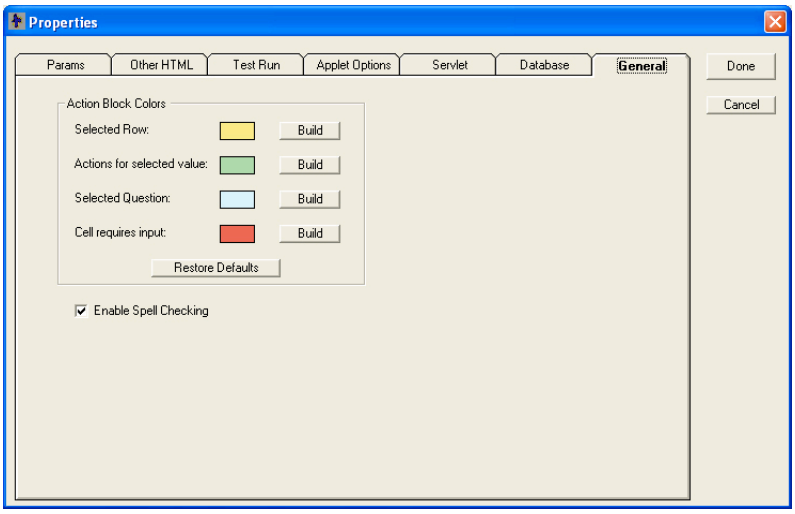
When running a system, the Corvid Runtime allows the user to step back either with the “Back” button in the applet window; or the browser “Back” button if using the Servlet Runtime. To support “Back”, the Corvid Runtime must maintain data about the session for each question that is asked. This takes up memory and increases resource utilization. Normally this is not a problem, but for very large systems, or when running on hardware with limited resources (e.g. a palmtop computer such as an iPAQ), the number of steps “Back” can be limited. This allows a system to reduce resources.



Buttons at the lower left of the Param tab allow the Undo Levels to be set to “Unlimited” or a specific value. If “Unlimited” is chosen, the system will support “Back” all the way to the first question. If a specific level is chosen, the “Back” button can be pressed that many times to step back in the run.

General Tab

The “General” tab on the Properties window allows setting the colors used in Action Blocks and enables/disables spell checking.



To change the colors, click the “Build” button next to the color and select a new color. The colors can be reset to the default colors by clicking the “Restore Defaults” button.

If you do not wish to use the spell checker, deselect the “Enable Spell Checking” checkbox. The spell checker only has an English dictionary so if you are working in other languages it is better to disable it

EXSYS
Capture Knowledge. Deliver Answers

EXSYS, Inc.
6301 Indian School Rd. NE, Suite 700
Albuquerque, NM 87110 USA
Tel: (505)888-9494, Fax: (505)888-9509, Email: info@exsys.com
www.exsys.com