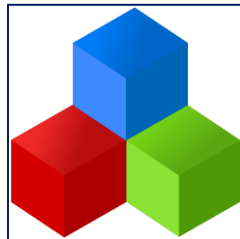


# ScenarioBuilder7

## User Guide



Protocol Independent  
Test Scripting Tool

<b>WELCOME TO SCENARIOBUILDER .....</b>	<b>6</b>
SCENARIOBUILDER FEATURES .....	6
<b>ABOUT THIS GUIDE.....</b>	<b>7</b>
<b>NRG GLOBAL SUPPORT .....</b>	<b>7</b>
<b>SCENARIOBUILDER INSTALLATION .....</b>	<b>7</b>
INSTALL SCENARIOBUILDER .....	8
END USER LICENSE AGREEMENT .....	15
<b>BASIC TERMINOLOGY .....</b>	<b>17</b>
<b>LAUNCH SCENARIOBUILDER .....</b>	<b>18</b>
NEW SCENARIO.....	19
<i>Start Application</i> .....	20
<i>Open Web App</i> .....	20
<i>Click on Application Icon</i> .....	20
<i>Blank</i> .....	21
USER INTERFACE .....	22
<i>Default Layout</i> .....	22
<i>View</i> .....	23
<i>ScenarioBuilder Tool Bar</i> .....	24
SCENARIO RECORDER .....	26
<i>Scenario Recorder Menu</i> .....	26
<i>Anchor Image</i> .....	27
<b>CREATING YOUR FIRST SCENARIO WITH SCENARIO RECORDER .....</b>	<b>29</b>
SCENARIO DESIGN BASICS .....	33
<i>Design Outline 1 – Airline Ticket Website</i> .....	33
<i>Design Outline 2 – Search for an Account in CRM</i> .....	35
<i>Design Outline 3 – Enter Hours in a Timesheet Application</i> .....	36
<b>CREATING YOUR FIRST SCENARIO WITH SCENARIOBUILDER .....</b>	<b>37</b>
<b>COMPONENTS .....</b>	<b>41</b>
DEFINE COMPONENTS .....	42
SAVE COMPONENTS .....	43
COPY A COMPONENT TO ANOTHER PROJECT.....	44
EXPORT AND IMPORT COMPONENTS .....	45
<i>Export Component</i> .....	45
<i>Import Component</i> .....	46
<b>WORKING WITH IMAGES.....</b>	<b>50</b>
THINK LIKE A REAL USER.....	50
<i>Image Actions Keep Scenarios Synchronized</i> .....	51
<i>Combine Image and Loop Actions to Page Down Until Image is Found</i> .....	52

<i>Wait for Image to Disappear Before Advancing</i> .....	54
<i>Allow for Image Variations</i> .....	56
<i>Image Appears Twice on the Desktop</i> .....	58
<b>WORKING WITH SCENARIO PARAMETERS</b> .....	<b>60</b>
ADDING PRE-DEFINED VARIABLES TO A SCENARIO.....	60
<i>Increment a Variable</i> .....	61
EXECUTION STATUS VARIABLES .....	62
USING A CSV FILE TO STORE VALUES FOR SCENARIO PARAMETERS .....	64
<i>Create A New Scenario Parameter CSV File</i> .....	64
<i>Edit an Existing Scenario Parameter CSV File</i> .....	65
<i>Attach a Scenario Parameter CSV File</i> .....	66
<i>Add Scenario Parameters</i> .....	67
<i>Last Line Used in Scenario Parameter CSV File</i> .....	67
<b>ACTION CONTEXT MENU</b> .....	<b>68</b>
<b>CONDITIONAL STATEMENTS</b> .....	<b>69</b>
"IF" ACTION .....	69
"IF" ACTION WITH EXECUTION STATUS VARIABLE .....	71
CASE/SWITCH.....	72
<b>ON FAILURE</b> .....	<b>75</b>
<b>SAMPLE ADVANCED SCENARIO</b> .....	<b>76</b>
<b>SCENARIO PROPERTIES</b> .....	<b>81</b>
<b>PROJECTS</b> .....	<b>82</b>
<b>OPTIONS</b> .....	<b>82</b>
WORK SPACE.....	83
<i>Change Active Project</i> .....	84
<i>Change Theme</i> .....	85
BASE .....	85
<i>Change Log Level</i> .....	86
GLOBAL HOTKEYS .....	87
SPLASH SCREEN SETTINGS .....	87
RECORDER .....	88
CREATE A PROJECT.....	89
<b>IMPORT AND EXPORT SCENARIOS</b> .....	<b>91</b>
<i>Export Scenario</i> .....	91
<i>Import Scenario</i> .....	92
<b>NESTING SCENARIOS</b> .....	<b>94</b>
WORKFLOWS.....	96
<b>FAVORITES</b> .....	<b>97</b>

<b>CLIPBOARD .....</b>	<b>98</b>
<b>ACTION GLOSSARY .....</b>	<b>100</b>
APPLICATION .....	100
<i>Start Application</i> .....	100
<i>Close Application</i> .....	100
<i>Begin Transaction</i> .....	101
<i>End Transaction</i> .....	101
<i>Execute Component</i> .....	101
<i>Define Component</i> .....	102
<i>Run Command Line</i> .....	103
<i>Open with File Association</i> .....	103
<i>Play Scenario</i> .....	104
IMAGES .....	104
<i>Click on Image</i> .....	104
<i>Find Image</i> .....	105
<i>Find Linked Images</i> .....	106
KEYBOARD .....	106
<i>Type Text</i> .....	106
<i>Function Keys</i> .....	107
MOUSE .....	107
<i>Mouse Click</i> .....	107
<i>Mouse Move and Click</i> .....	108
<i>Mouse Move</i> .....	108
WINDOW .....	108
<i>Window Actions</i> .....	109
<i>Move Window</i> .....	109
<i>Resize Window</i> .....	110
<i>All Windows Action</i> .....	111
CONDITIONALS AND VARIABLES .....	111
<i>If</i> .....	112
<i>Loop</i> .....	113
<i>Switch</i> .....	113
<i>Case</i> .....	113
<i>Increment a Variable</i> .....	113
<i>Set a Variable</i> .....	114
<i>Read Text into a Variable</i> .....	114
<i>Rendezvous Point</i> .....	115
<i>Exit</i> .....	115
<i>Label</i> .....	115
<i>On Failure</i> .....	116
<i>Resume</i> .....	116
<i>Skip To</i> .....	116
<i>Raise Failure</i> .....	116
MISCELLANEOUS .....	116

<i>Timer</i> .....	116
<i>Comment</i> .....	117
<i>Wait</i> .....	117
<i>CallDll</i> .....	117
<i>Line Filler</i> .....	117
<i>Read Serial Port</i> .....	117
<i>Write Serial Port</i> .....	118
<b>SCENARIO RECORDER GLOSSARY</b> .....	<b>118</b>
CAPTURE IMAGE .....	118
ACTIONS .....	119
UNDO .....	120
TREE VIEW .....	120
PREFERENCES .....	121
<b>APPENDIX A – USE SCENARIOS WITH APPSWATCH</b> .....	<b>124</b>
APPSWATCH ARCHITECTURE .....	124
SEND A SCENARIO TO APPSWATCH BASE .....	124
ADD NEW APPLICATION MONITOR .....	125
<b>APPENDIX B – USE SCENARIOS WITH APPLOADER</b> .....	<b>127</b>
SEND A SCENARIO TO APPLOADER CONTROLLER .....	127
MANAGE INJECTORS .....	129
RUN THE LOAD TEST .....	130
<b>INDEX</b> .....	<b>132</b>

## WELCOME TO SCENARIOBUILDER

ScenarioBuilder from NRG Global is an easy-to-use tool that simplifies test development. Used in conjunction with AppLoader or AppsWatch, ScenarioBuilder makes it easy to create a wide range of test scenarios for your load testing or application monitoring needs.

With ScenarioBuilder's intuitive point and click user interface, users of all skill levels can create any sequence of end user actions, which can be played back in test scenarios. No coding or complex scripting languages are required. Scenario Recorder captures keystrokes in real time, further simplifying the process. As a result, you can develop even the most complex test scenarios in a fraction of the time required with other load testing and application monitoring solutions.

ScenarioBuilder also features a handy playback tool that allows you to verify your test scenario right on your desktop. You can test your scenario along the way, as well as verify the final scenario before sending it to AppsWatch Base or AppLoader Controller for execution in the test environment.

## SCENARIOBUILDER FEATURES

With ScenarioBuilder, you can emulate real user activities, including mouse events, keyboard inputs, and more. Here's a sampling of the user *Actions* you can create in a test Scenario:

- Type text, including function keys
- Mouse moves and clicks
- Move and resize windows
- Find and click on an image (for Windows objects that don't have a "handle", ScenarioBuilder's image Actions provide the flexibility to navigate to virtually anywhere on the desktop)
- Define a specific image "Search Area" on the desktop

In addition to basic user Actions, ScenarioBuilder includes advanced conditional statements and variables, so you can easily create complex scenarios without any scripting:

- If/Then/Else statements
- Loops
- Case/Switch statements
- Variables retrieved from a CSV file allow you to change values with each scenario execution
- Determine Actions to take if your scenario is interrupted by timeouts, pop-ups, or other failures

ScenarioBuilder's tools and options provide flexible, user friendly functionality

- Create *Components* to encapsulate Action sequences for reuse throughout the current, or in future scenarios
- Record Actions in real time with Scenario Recorder
- Edit and update recorded scenarios in either "Graphical" or "Tree" views

## ABOUT THIS GUIDE

Designed for both the novice and the experienced user, this guide will introduce you to ScenarioBuilder's features and walk you through all the steps involved in creating test scenarios.

First time users should review the [Getting Started — First Time Users](#) section to become familiar with ScenarioBuilder's user interface, tool bars, and layout, as well as learn how to create a basic test scenario.

The [Creating Scenarios – Best Practices](#) section contains details and best practices to help you develop more complex test scenarios.

The [Index A](#) at the end of this document provides a glossary of ScenarioBuilder Actions with brief descriptions.

This guide covers building test scenarios, playing them back for verification, and sending test scenarios to AppsWatch Base or AppLoader Controller. For more details on executing test scenarios from NRG Global AppsWatch and AppLoader, and gathering and interpreting results from these applications, please refer to the AppLoader or AppsWatch User Guides.

## NRG GLOBAL SUPPORT

Please visit <http://www.nrgglobal.com/resources/training-videos> to view our ScenarioBuilder video training series.

ScenarioBuilder provides a library of sample scenarios which can be found in the **Help** menu under **Sample Scenarios**.

Contact our technical support team:

By phone: (213) 234-7550; (888) 624-4447 (toll-free)

By email: [support@NRGglobal.com](mailto:support@NRGglobal.com)

## SCENARIOBUILDER INSTALLATION

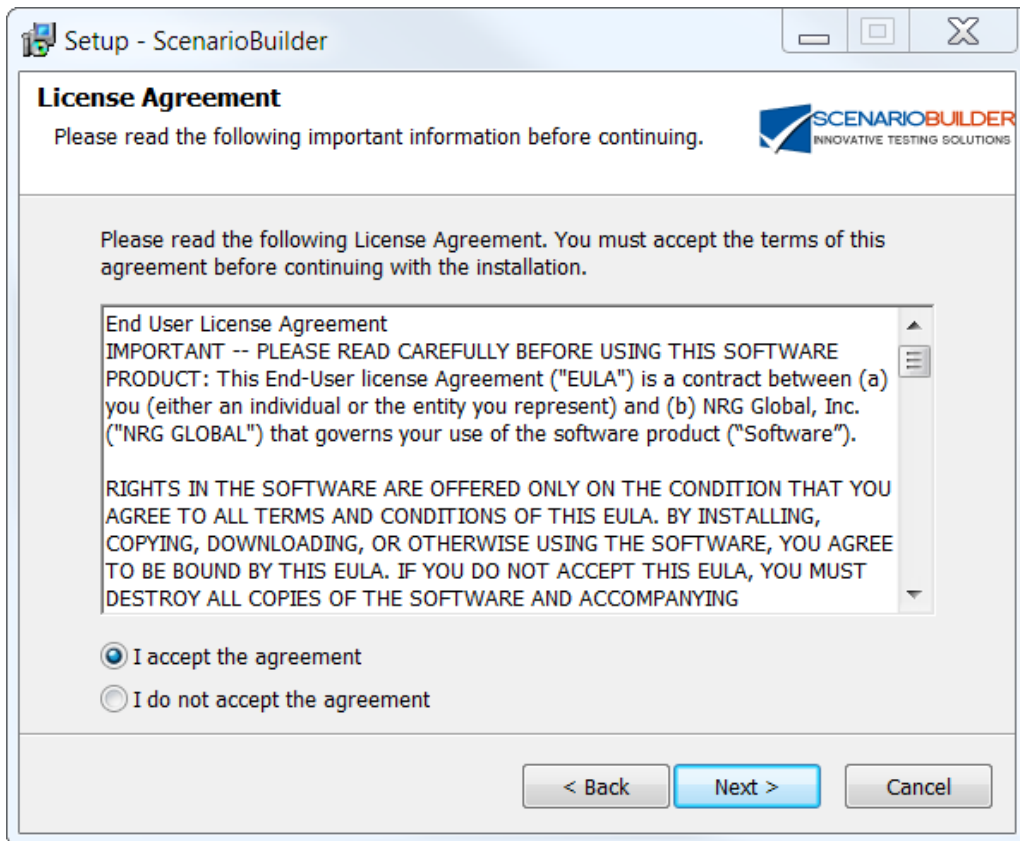
Start by downloading the ScenarioBuilder installation package from the *Product Downloads* page on the NRG Global website. ScenarioBuilder's installation package can be accessed by clicking the <http://nrgglobal.com/viewcategory/3> link. Click the ScenarioBuilder link to save the *scenariobuilder\_setup.zip* file to your Windows PC and extract the contents of the file to a new folder.

## INSTALL SCENARIOBUILDER

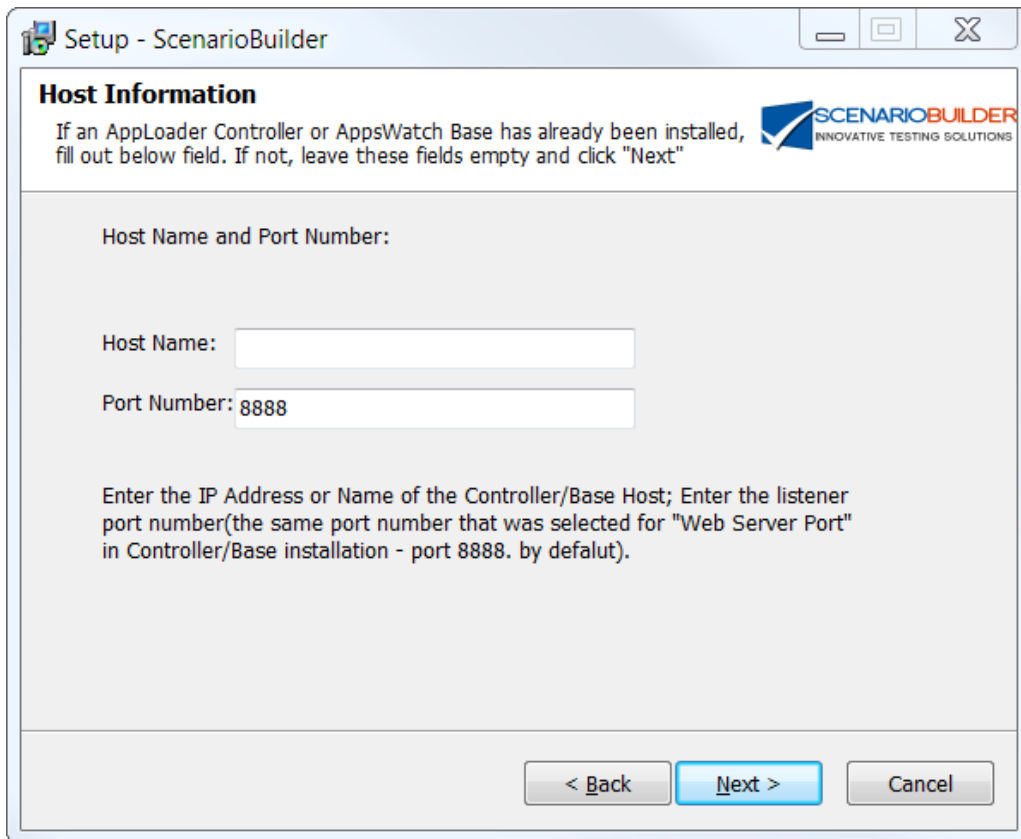
Run the *setup.exe* file in the ScenarioBuilder folder.



Click "Next" to advance to the "End User License Agreement" (EULA).



Review the terms of the EULA and click "I Agree" to continue.



**Setup - ScenarioBuilder**

**Host Information**

If an AppLoader Controller or AppsWatch Base has already been installed, fill out below field. If not, leave these fields empty and click "Next"

Host Name and Port Number:

Host Name:

Port Number:

Enter the IP Address or Name of the Controller/Base Host; Enter the listener port number(the same port number that was selected for "Web Server Port" in Controller/Base installation - port 8888. by defalut).

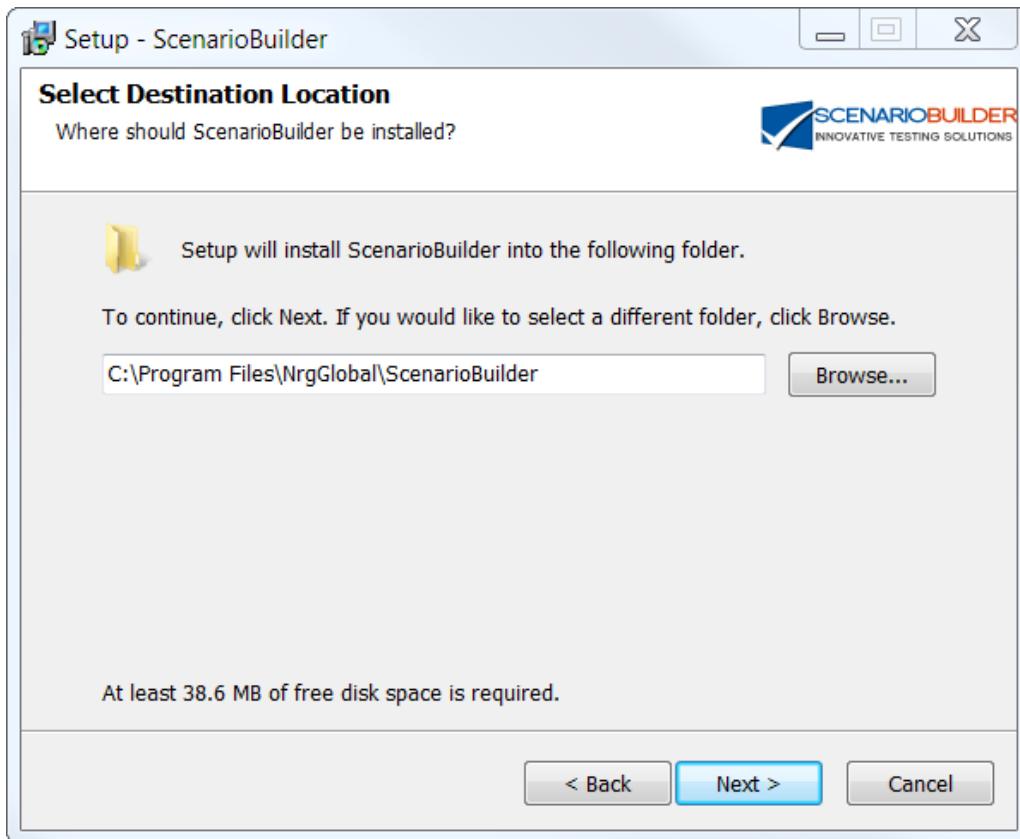
< Back   Next >   Cancel

Enter the IP address of the server hosting the Controller (if using with AppLoader) or Base (if using with AppsWatch) in the "Host Name" field.

Enter the "Port Number" address – use the same port that was assigned for "Web Server Port" in the Controller/Base installation.

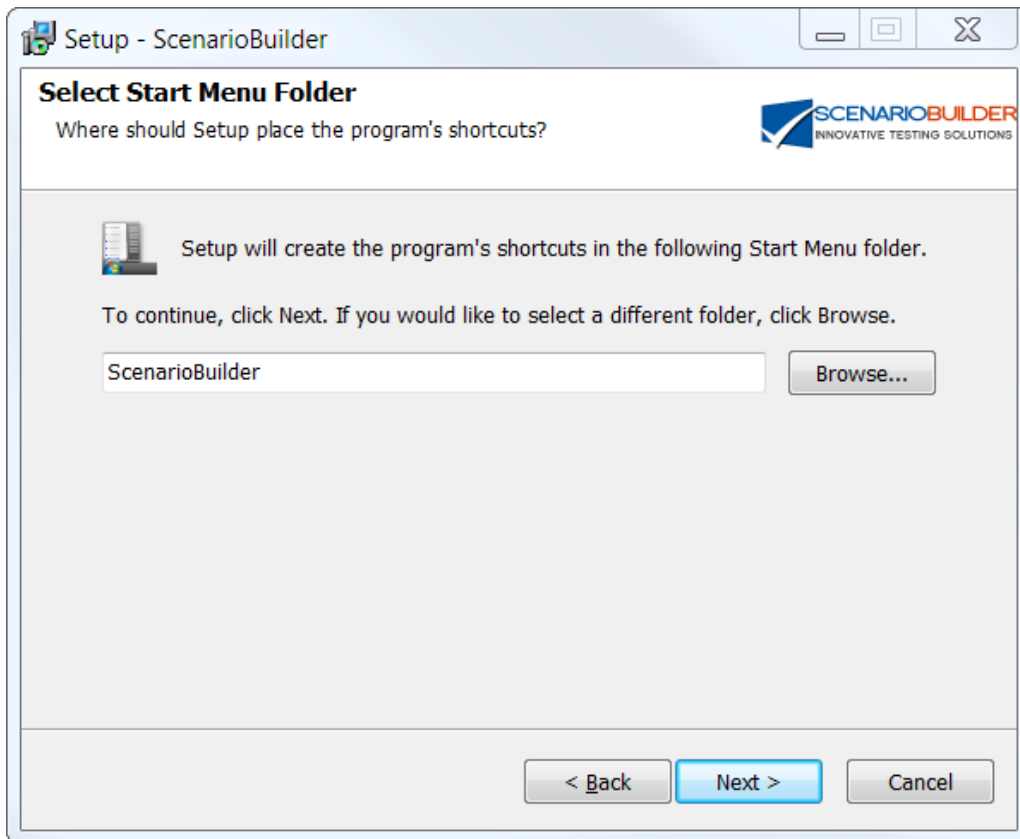
Click "Next."

*Note: If upgrading from an older version or reinstalling the current version of ScenarioBuilder, this window will not appear – prior settings will be used.*



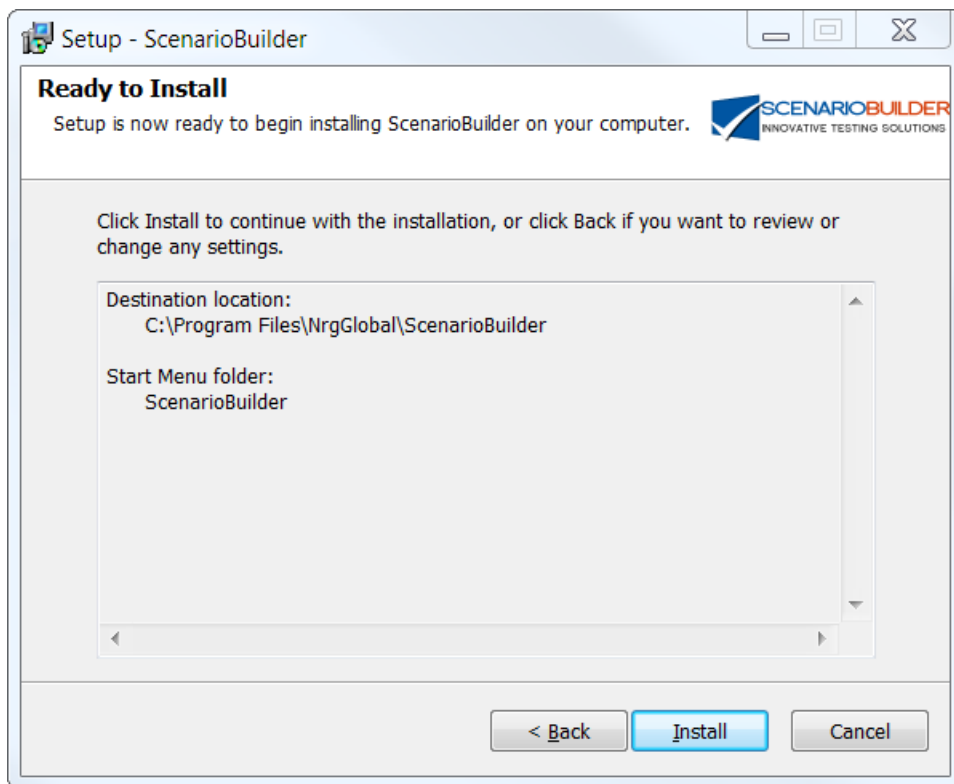
Select the destination folder for the ScenarioBuilder program installation. Click “Next” to accept the default location or “Browse” to an alternate location, then click “Next.”

*Note: If upgrading from an older version or reinstalling the current version of ScenarioBuilder, this window will not appear – prior settings will be used.*

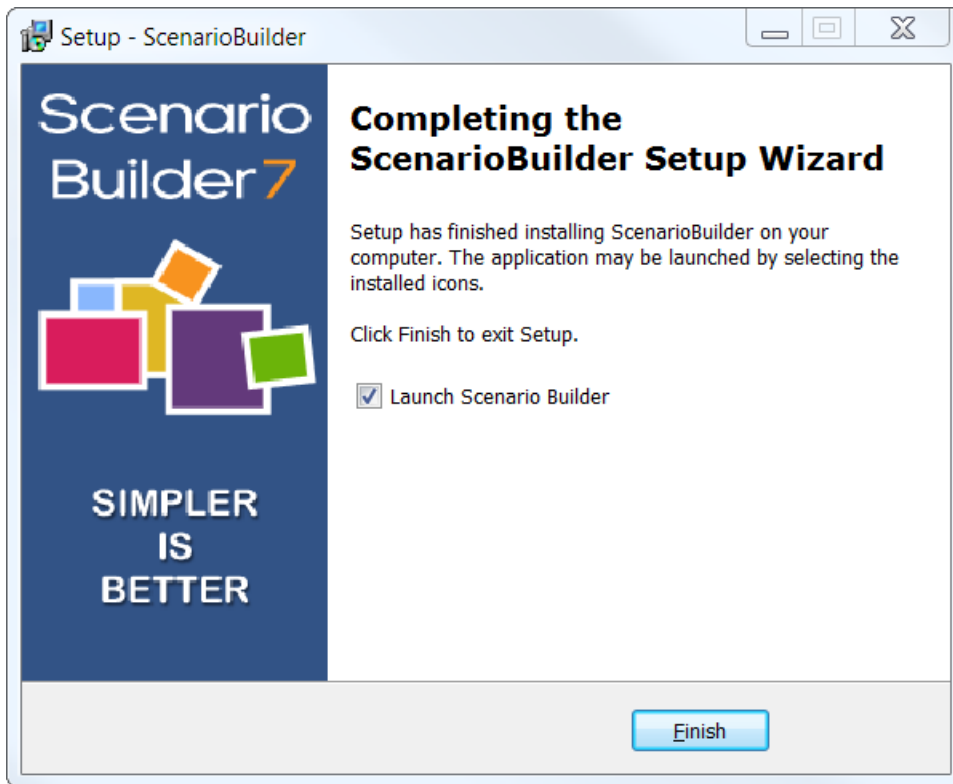


Select the destination folder for the ScenarioBuilder program shortcuts. Click “Next” to accept the default location or “Browse” to an alternate location, then click “Next.”

*Note: If upgrading from an older version or reinstalling the current version of ScenarioBuilder, this window will not appear – prior settings will be used.*



Click "Install" to initiate the installation process.



Click “Finish” to complete the ScenarioBuilder installation.

## END USER LICENSE AGREEMENT

**IMPORTANT -- PLEASE READ CAREFULLY BEFORE USING THIS SOFTWARE PRODUCT:** This End-User license Agreement ("EULA") is a contract between (a) you (either an individual or the entity you represent) and (b) NRG Global, Inc. ("NRG GLOBAL") that governs your use of the software product ("Software").

RIGHTS IN THE SOFTWARE ARE OFFERED ONLY ON THE CONDITION THAT YOU AGREE TO ALL TERMS AND CONDITIONS OF THIS EULA. BY INSTALLING, COPYING, DOWNLOADING, OR OTHERWISE USING THE SOFTWARE, YOU AGREE TO BE BOUND BY THIS EULA. IF YOU DO NOT ACCEPT THIS EULA, YOU MUST DESTROY ALL COPIES OF THE SOFTWARE AND ACCOMPANYING DOCUMENTATION WITHIN 1 DAY OF PURCHASE FOR A REFUND FROM NRG GLOBAL.

1. **THIRD PARTY SOFTWARE.** The Software may include, in addition to NRG GLOBAL proprietary software ("NRG GLOBAL Software"), software under licenses from third parties ("Third Party Software" and "Third Party License"). Any Third Party Software is licensed to you subject to the terms and conditions of the corresponding Third Party License. Generally, the Third Party License is located in a file such as license.txt or read.me file; you should contact NRG GLOBAL support if you cannot find any Third Party License.

2. **LICENSE RIGHTS.** You will have the following rights provided you comply with all terms and conditions of this EULA: a. **Use.** NRG GLOBAL grants you a license to Use one copy of the NRG GLOBAL Software. "Use" means installing, copying, storing, loading, executing, displaying, or otherwise using the NRG GLOBAL Software. You may not modify the NRG GLOBAL Software or disable any licensing or control feature of the NRG GLOBAL Software. Additional restrictions on Use may appear in the User Documentation. You may not separate component parts of the NRG GLOBAL Software for Use. You do not have the right to distribute the NRG GLOBAL Software. b. **Copying.** You may copy the NRG GLOBAL software for back-up and archival purposes, provided any copy contains all the original software's proprietary notices.

3. **UPGRADES.** To Use NRG GLOBAL Software provided by NRG GLOBAL as an upgrade, update, or supplement (collectively "Upgrade"), you must first be licensed for the original NRG GLOBAL Software identified by NRG GLOBAL as eligible for the Upgrade. To the extent the Upgrade supersedes the original NRG GLOBAL Software, you may no longer use such NRG GLOBAL Software. This EULA applies to each Upgrade unless NRG GLOBAL provides other terms with the Upgrade. In case of a conflict between this EULA and such other terms, the other terms will prevail.

4. **TRANSFER.** a. **Third Party Transfer.** The initial end user of the NRG GLOBAL Software may make a one-time transfer of the NRG GLOBAL Software to another end user. Any transfer will include all component parts, media, User Documentation, this EULA, and if applicable, the Certificate of Authenticity. The transfer may not be an indirect transfer, such as a consignment. Prior to the transfer, the end user receiving the transferred Software will agree to this EULA. Upon transfer of the NRG GLOBAL Software, your license is automatically terminated. b. **Restrictions.** You may not rent, lease or lend the NRG GLOBAL Software or Use the NRG GLOBAL Software for commercial timesharing or bureau use. You may not sublicense, assign or otherwise transfer the NRG GLOBAL Software except as expressly provided in this EULA.

5. PROPRIETARY RIGHTS. All ownership and intellectual property rights in the Software and User Documentation are owned by NRG GLOBAL and/or its suppliers and are protected by law, including applicable copyright, trade secret, patent, and trademark laws. You will not remove any product identification, copyright notice, or proprietary restriction from the Software.

6. LIMITATION ON REVERSE ENGINEERING. You may not reverse engineer, decompile, or disassemble the NRG Global Software, except and only to the extent that the right to do so is allowed under applicable law.

7. CONSENT TO USE OF DATA. NRG Global and its affiliates may collect and use technical information you provide in relation to (i) your Use of the Software or the NRG GLOBAL Product, or (ii) the provision of support services related to the Software or the NRG GLOBAL Product. All such information will be subject to NRG GLOBAL's privacy policy. NRG GLOBAL will not use such information in a form that personally identifies you except to the extent necessary to enhance your Use or provide support services.

8. LIMITATION OF LIABILITY. Under no circumstances shall NRG GLOBAL or its suppliers or resellers be liable to licensee or any other person for any special, incidental, indirect, or consequential damages, including damages for loss of goodwill, lost profits, work stoppage, business interruption, personal injury, computer failure, loss of privacy, or any and all commercial damages or losses, even if NRG GLOBAL or any supplier has been advised of the possibility of such damages. Further, in no event shall NRG GLOBAL's liability under any provision of this agreement exceed the license fee paid to NRG GLOBAL. Some states or other jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

9. U.S. GOVERNMENT RESTRICTED RIGHTS AND COMPLIANCE WITH EXPORT LAWS. This software is provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause of DFARS 252.227-7013 or subparagraphs (c)(i) and (2) of the Commercial Computer Software-Restricted Rights at 48 CFR 52.227-19, as applicable. You will comply with all laws, rules, and regulations (i) applicable to the export or import of the Software, or (ii) restricting the Use of the Software.

10. RESERVATION OF RIGHTS. NRG Global and its suppliers reserve all rights not expressly granted to you in this EULA.

## BASIC TERMINOLOGY

**Scenario:** A scenario is an XML file created with ScenarioBuilder. A scenario contains Actions that emulate the activity of a user interacting with an application. Scenarios can be called in other scenarios (known as [nesting](#)) with the “Play Scenario” Action.

**Action:** An Action is a step in a scenario that represents a single event (such as a mouse click or an application launch). Actions are located in a pane on the left side of the ScenarioBuilder window, and can be added to scenarios by double-clicking or dragging into the Scenario Window.

**Component:** A Component is a user-defined shell within a scenario that contains a single Action or a series of Actions. Think of Components as containers for Actions related to a specific task in the overall process. Breaking up scenarios into Components makes for easier scenario maintenance, and can be helpful in conceptualizing a scenario design. Components are self-contained units that can be called from any scenario within a Project. Additionally, Components can be exported and imported to and from Projects. If a modification is made to a Component that is being used by multiple scenarios, the Component is updated for all scenarios.

**Transaction:** A Transaction is a section of a scenario that is marked for the purpose of measuring the response time of that section. For example, you can define a Transaction to be “login to CRM” where the beginning of the Transaction is defined when the CRM application is launched, and the end of the Transaction is defined when the CRM interface finishes loading on the screen. Many Actions may occur within a Transaction (such as finding the executable to launch the application, launching the executable, waiting for the login screen, typing in the login name and password, waiting for the application to load completely, etc.).

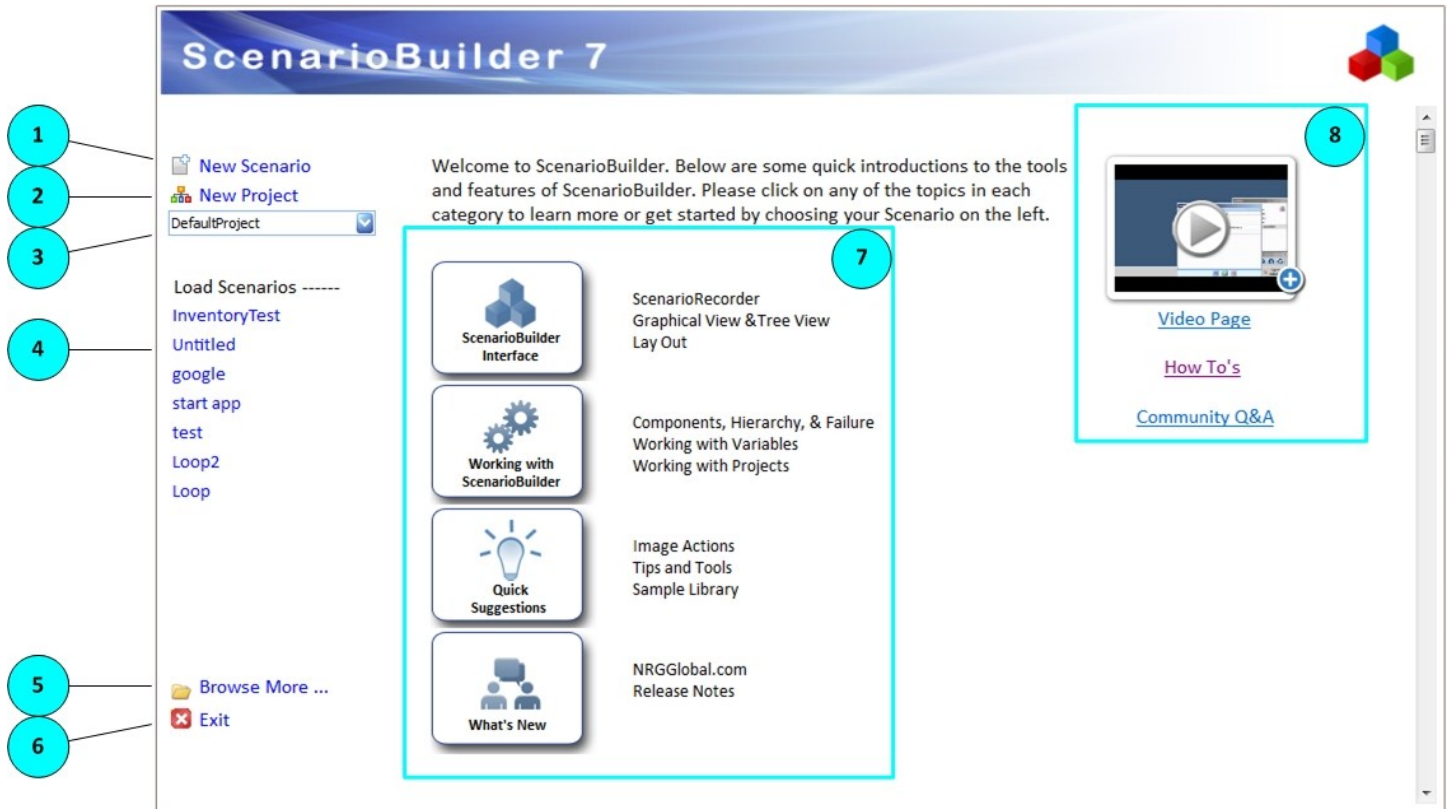
**Project:** A Project is a folder that contains the elements of your scenarios (Images, Variables, scenario XML files, Components and Documents). Upon installation, ScenarioBuilder creates the “Default Project” folder, where scenarios will be stored *unless* a new Project is created and “Switched To”. Creating a Project for each business application being tested not only keeps test cases organized, but allows you to share resources among related scenarios. For example, imagine a separate scenario needs to be created to test various features of an application. Each of these features uses a common icon. The time-saving benefits of sharing a single image for each scenario are two-fold; (1) the image can simply be “loaded” rather than “captured” each subsequent time it’s needed; (2) in the event that the “Submit” button is modified or changed in any way, updating the image once will update it for all of the scenarios within the Project. To create a Project, click **File→Project Manager→New Project**. A Project folder will be created, with subfolders for Images, Variables, scenario XML files, Components and documents.

**On Failure:** The “On Failure” section of a scenario determines what Action(s) to take when a scenario or Component fails. For example, if your application crashes during a test, “On Failure” Actions could shut down and restart the application and attempt to resume the scenario where the crash occurred. A “Raise Failure” Action used in conjunction with an “On Failure” will mark and report the crash as a failure. If “Raise Failure” is omitted, the scenario will still return a “success” status.

**Response Time:** Response time is the amount of time taken to execute a Scenario, Component, Transaction or step within a scenario.



## LAUNCH SCENARIOBUILDER

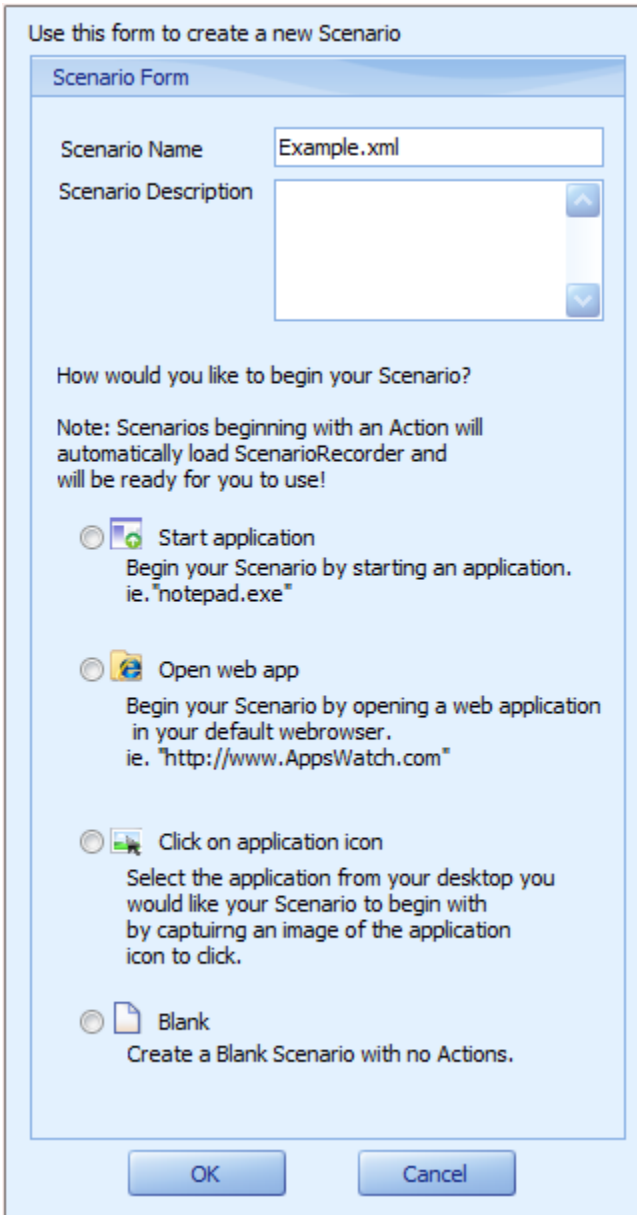
The ScenarioBuilder “Splash Screen” greets you upon launching the program. Here, you may create a New Scenario, open an existing Scenario, or familiarize yourself with the common Actions used in ScenarioBuilder;



- 1) **New Scenario:** Click to launch the “New Scenario Properties” form which guides you through starting a new scenario.
- 2) **New Project:** Click to create a new Project.
- 3) **Switch Project:** Click to switch to another Project.
- 4) **Load Scenarios:** Choose an existing scenario to edit. The list is populated with the most recent scenarios from the current project. Choose an alternate Project from the drop-down box and view scenarios from that Project.
- 5) **Browse More:** Click to get the complete list of scenarios in the selected Project.
- 6) **Exit:** Click to exit ScenarioBuilder.
- 7) **Quick Introductions:** A quick reference guide to the features of ScenarioBuilder. Use the scroll bar or click on the links to browse through each of the features.
- 8) **Help & Support Guide:** View video demonstrations, “How To” blogs and community support forums.

## NEW SCENARIO

When you click the “New Scenario”  [New Scenario](#) link from the Splash Screen, click the “New Scenario”  icon from the ScenarioBuilder tool bar, or select “New” from the ScenarioBuilder “File” menu, the “New Scenario Properties” form displays. The form will get you started with your new Scenario;



Use this form to create a new Scenario


**Scenario Form**


Scenario Name


Scenario Description


How would you like to begin your Scenario?

Note: Scenarios beginning with an Action will automatically load ScenarioRecorder and will be ready for you to use!

☐  Start application  
Begin your Scenario by starting an application.  
ie. "notepad.exe"

☐  Open web app  
Begin your Scenario by opening a web application  
in your default webbrowser.  
ie. "http://www.AppsWatch.com"

☐  Click on application icon  
Select the application from your desktop you  
would like your Scenario to begin with  
by capturing an image of the application  
icon to click.

☐  Blank  
Create a Blank Scenario with no Actions.

OK Cancel


**Scenario Name:** Enter a meaningful name for the scenario.

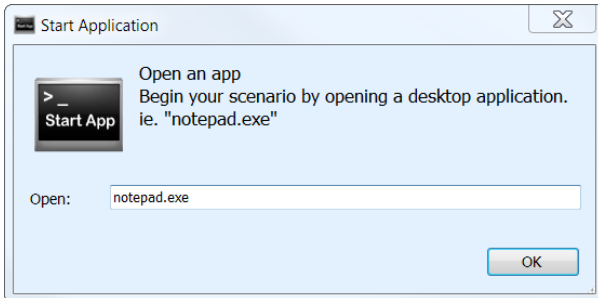
**Scenario Description:** Add an optional description to help identify the scenario.

*Select one of the four ways for your scenario to begin.*

---

## START APPLICATION

**Start Application**  - To start an installed application such as notepad.exe or iexplorer.exe. This choice opens ScenarioBuilder in [Scenario Recorder](#) mode. Begin by completing the form;




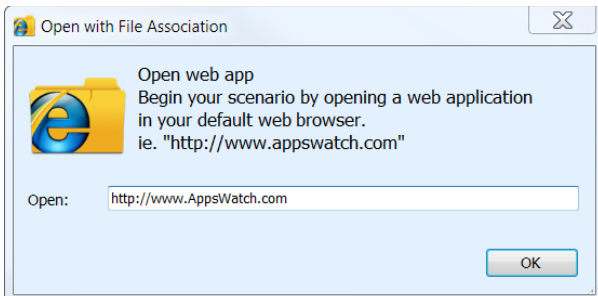
**Open:** Enter the name and extension of the executable file. Provide the full path unless the file resides in the Windows path. Click “OK”. Application will open;

Scenario Recorder will launch and prompt you to capture an anchor image.

---

## OPEN WEB APP

**Open Web App**  - To launch a web application with your default browser. This choice opens ScenarioBuilder in [Scenario Recorder](#) mode. Begin by completing the form;




**Open:** Provide the complete URL address of the web application (e.g. <http://www.AppsWatch.com>). Click “OK”. Web application will open;

Scenario Recorder will launch and prompt you to capture an anchor image.

---

## CLICK ON APPLICATION ICON

**Click on Application Icon**  - To capture an image for the mouse to click on (e.g. desktop shortcut to launch an application). This choice opens ScenarioBuilder in [Scenario Recorder](#) mode, first by initiating the screen capture tool;

The screen capture tool opens with a default sized selection box. Locate the selection box over the desired image and click;




To alter selection box size, click [Ctrl] key to change mode. Click [Ctrl] key (again), then hold down the left mouse button and drag your cursor around the desired image, then release the left mouse button;



Scenario Recorder will launch.

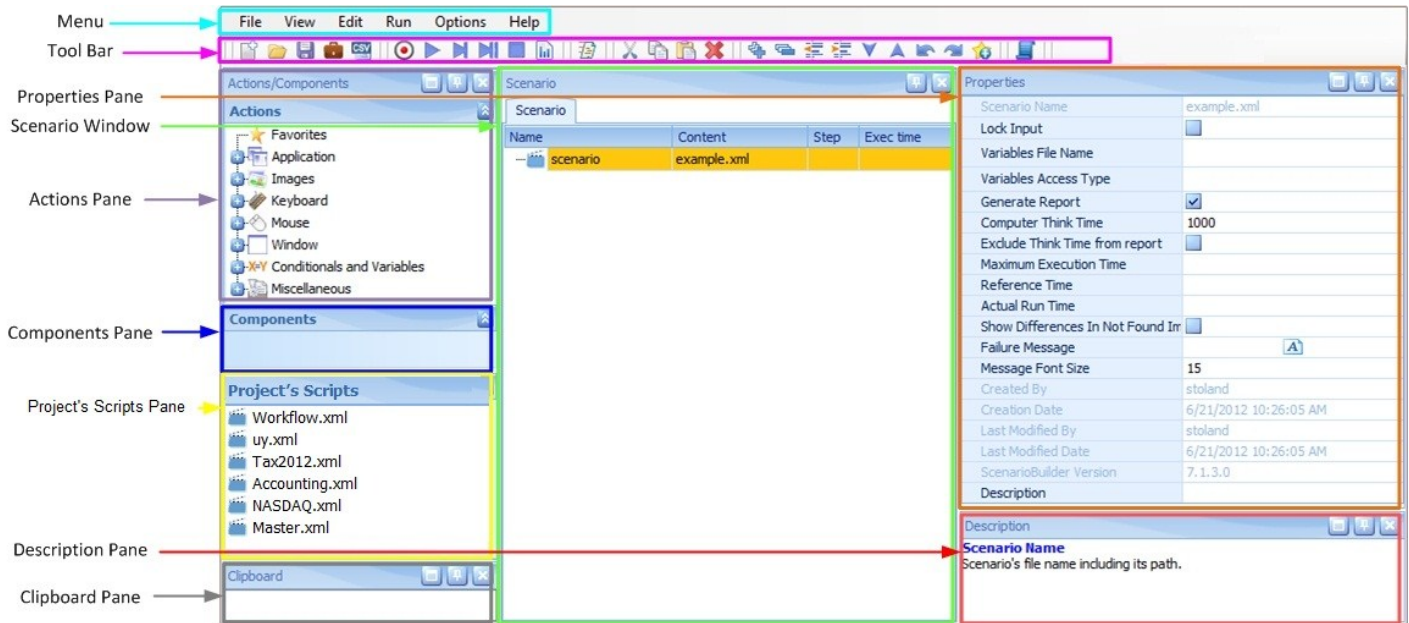
## BLANK

**Blank**  - This choice simply starts an empty scenario with no Actions. ScenarioBuilder opens to the default layout.

## USER INTERFACE

ScenarioBuilder's user interface can be adjusted to meet your personal preferences. The following screen shows the default layout. Keep this default layout, or move/resize the panes however you like. If you prefer that the Properties pane is right next to the Actions pane (to reduce cursor movement when updating properties), simply drag and drop the Properties pane across the screen.

### DEFAULT LAYOUT



**Menu:** ScenarioBuilder's menu options are grouped under the *File*, *View*, *Edit*, *Run*, *Options* and *Help* menus.



**Tool Bar:** ScenarioBuilder provides an assortment of commonly used menu options in the Tool Bar for quick access.

**Scenario Window:** The Scenario Window is the heart of a scenario. It contains the step by step series of Actions in the order in which they will be executed. Add Actions to the scenario by either dragging them into the Scenario Window or simply double-clicking on them (add Component and Component Templates to the scenario by double-clicking on them). Actions can be relocated within the Scenario Window by clicking and dragging them, or by highlighting them and clicking the "Up" ▲ or "Down" ▼ icons in the tool bar. For each Component or Component Template you add to your scenario, a tab for that Component is automatically added to the Scenario Window. The Scenario Window shows, by default, "Action Name" and "Content", however the view can be customized to include "Step" and "Execution Time".

**Actions Pane\*:** Actions are the building blocks of scenarios. An Action is a step in a scenario that represents a single event (such as a mouse click or an application launch). ScenarioBuilder includes a comprehensive array of end-user Actions to facilitate the creation of virtually any scenario. This pane contains all of these Actions. Use the "+" or "-" symbols to expand or collapse any of the Action groups.

**Components Pane\***: Saved Components in the active Project will be listed in the Components pane. Double-click a Component to add to the current scenario.

**Project's Scripts Pane\***: ScenarioBuilder displays all of the scenarios in the current Project in this pane. Double-click on a scenario from this pane to add it to the current scenario via the "Play Scenario" Action.

*\* The Actions, Components and Project's Scenarios panes are all located inside the Actions/Components window. **Only one of the three panes can be expanded at one time.** Use the "expand"  and "collapse"  buttons to reveal the desired pane.*

**Properties Pane**: Every Scenario, Component and Action has an associated set of properties. These properties are displayed in the Properties pane when the item is highlighted in the Scenario Window. Property values can be edited in the Properties pane.

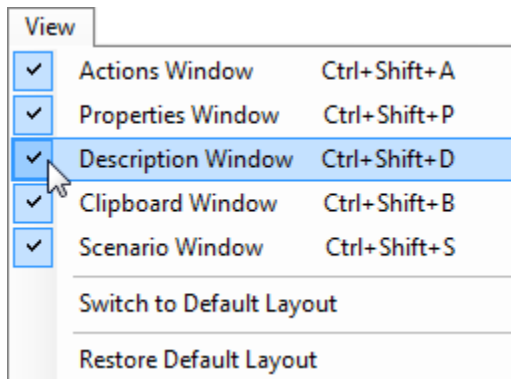
**Clipboard**: Copy an Action or a series of Actions to the Clipboard to be recalled later in the current scenario. Actions copied to the Clipboard retain their Property settings. Use the Clipboard when Actions with the same properties are repeated in different places within the scenario. Highlight the Action(s) and right-click. Choose "Copy" from the menu and the item(s) will be added to the Clipboard. Then double-click on the item in the Clipboard to add it to the scenario. Note that the Clipboard is cleared whenever ScenarioBuilder is closed.

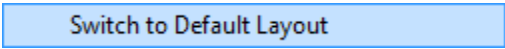
**Description**: In the Description pane, ScenarioBuilder offers a brief description and/or tip for the highlighted Action or property field.

---

## VIEW

You may hide or show panes by selecting or deselecting them in the "View" menu;



If at any point you go too far with your personalization and would like to return to the default layout, select "Switch to Default Layout"  from the "View" menu and ScenarioBuilder will revert to the default layout.






## SCENARIOBUILDER TOOL BAR

ScenarioBuilder provides an assortment of commonly used menu options in the Tool Bar for quick access;







Below is a directory and brief description of the tool bar options;

<u>Icon</u>	<u>Name</u>	<u>Description</u>
	<b>New Scenario</b>	Create a new scenario.
	<b>Open Scenario</b>	Browse the current Project to open an existing scenario.
	<b>Save Scenario</b>	Save the scenario and Components.
	<b>Project Documents</b>	Open the current Project's "Document" folder.
	<b>Scenario Parameters</b>	Create a CSV (comma separated values) file containing values for Scenario Parameters, or open and modify an existing CSV file.
	<b>Scenario Recorder</b>	Launch Scenario Recorder.
	<b>Play Scenario from Start</b>	Play the entire scenario.
	<b>Play from Position</b>	Play the scenario starting at the highlighted step.
	<b>Play Selected Step(s)</b>	Play the highlighted step(s) only.
	<b>Stop Scenario Playback</b>	Halt the playback of the scenario.
	<b>Execution Report</b>	View the most recent Execution Report
	<b>Send Scenario to Controller</b>	Send the scenario to either the AppsWatch Base or AppLoader Controller (command will not execute if "Reference Time" has not been sent).

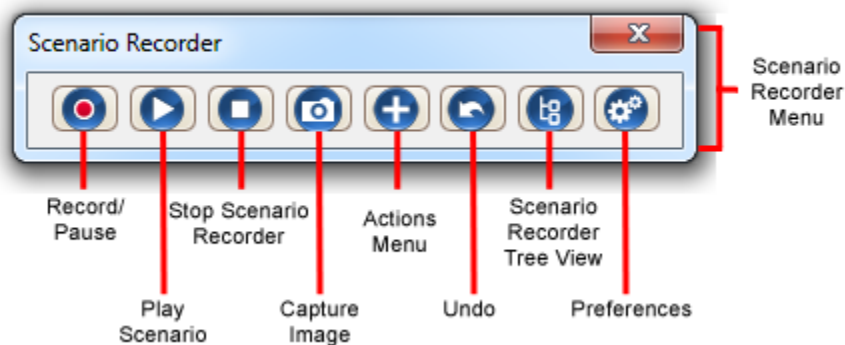
	<b>Cut Selection</b>	Cut (copy to clipboard, and remove) the highlighted step(s).
	<b>Copy Selection to Clipboard</b>	Copy the highlighted step(s) to the clipboard.
	<b>Paste Selection from Clipboard</b>	Paste the latest “copied” item from the clipboard to the scenario.
	<b>Delete Selection</b>	Delete the highlighted step(s).
	<b>Move Selection to the Left</b>	Move a child Action left, toward the parent level.
	<b>Move Selection to the Right</b>	Move an Action under the preceding Action, making it a child of the preceding. The child Action will only execute if the parent Action is successful.
	<b>Move Selection Down</b>	Move the highlighted Action down in the scenario.
	<b>Move Selection Up</b>	Move the highlighted Action up in the scenario.
	<b>Undo Last Action</b>	Reverse the last change made to the Scenario, restoring it to an earlier state. Undo can be used repeatedly to erase a series of changes.
	<b>Redo Last Action</b>	Reverse the effects of the “Undo” command. Redo can be used for each “Undo” command performed.
	<b>Add to Favorites</b>	Add an Action to the Favorites folder to be called later in the current scenario or in a future scenario (from the same Project). Actions saved to Favorites include child Actions, and they retain their Property settings.
	<b>Group Actions</b>	Group redundant Actions under a loop, for organizational purposes.



## SCENARIO RECORDER

Scenario Recorder helps you create your test scenarios fast! In *record* mode, keystrokes, mouse clicks and mouse moves are captured as they occur. Additionally, the Scenario Recorder menu provides quick access to many of the commonly used Actions in ScenarioBuilder, while remaining in record mode.



Beginning a scenario with the “Start Application” , “Open Web App” , or “Click on Application Icon”  methods brings you immediately into record mode. If in ScenarioBuilder standard view, clicking the “Scenario Recorder”  icon in the tool bar will also initiate the recorder.








Upon launching Scenario Recorder, ScenarioBuilder hides, and the recorder menu opens in the lower right corner of the desktop;



To close Scenario Recorder, simply “Stop”  recording or “Exit”  and ScenarioBuilder reverts to the default layout view.


## SCENARIO RECORDER MENU

<u>Icon</u>	<u>Name</u>	<u>Description</u>
	<b>Pause Recording</b>	Pause the recording session, but keep Scenario Recorder open.
	<b>Start Recording</b>	Record keystrokes fired from your keyboard.

	<b>Play Scenario</b>	Play the entire scenario from Scenario Recorder.
	<b>Stop Recording</b>	Stop the recording session. Scenario Recorder closes and ScenarioBuilder displays. Recorded Actions appear in the Scenario Window where they can be edited.
	<b>Capture Image</b>	Initiate all “Image” Actions while in Scenario Recorder mode.
	<b>Scenario Recorder “Actions” Menu</b>	Initiate basic “Application” Actions while remaining in Scenario Recorder mode.
	<b>Undo</b>	Reverse the last change made to the Scenario, restoring it to an earlier state. Undo can be used repeatedly to erase a series of changes.
	<b>Scenario Recorder “Tree” View</b>	View/Hide the Scenario Window while remaining in Scenario Recorder mode.
	<b>Scenario Recorder “Preferences”</b>	Manage hotkeys that trigger Actions or menu options in Scenario Recorder. Also, configure Scenario Recorder “Image” Action <i>Naming</i> and <i>Mouse Action</i> conventions.

## ANCHOR IMAGE

Upon launching, Scenario Recorder alerts you to capture an anchor image before proceeding!



**Click this button to capture an anchor image!**

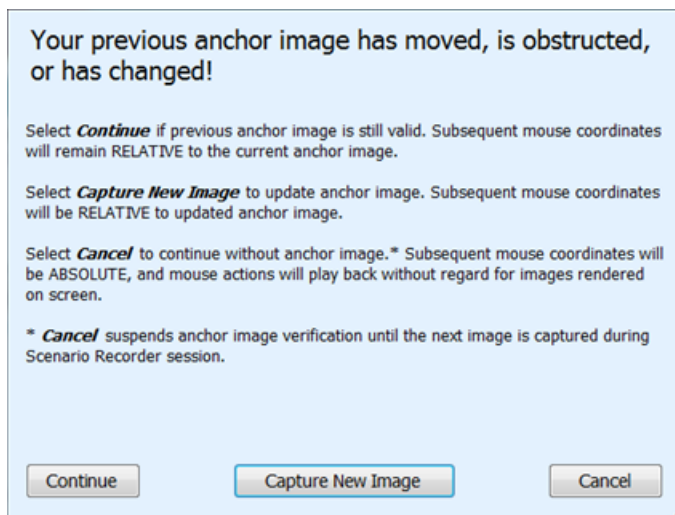
An anchor image provides a **RELATIVE** point-of-reference for recorded mouse move and mouse click coordinates, and ensures that mouse actions do not occur until anchor image has loaded. If an anchor image is not used, mouse coordinates are **ABSOLUTE**, and mouse actions will play back without regard for images rendered on screen.

The anchor image serves two purposes:

1. It tells ScenarioBuilder to wait and “find” this image before advancing – thereby keeping scenario engaged, or *synchronized* with the screen;
2. It provides a relative point-of-reference for the subsequent mouse clicks which will occur on the current page;

You may continue without using an anchor image. However, it should be noted that if you do not use an anchor image, subsequent mouse clicks will occur based on absolute coordinates (not relative to any image on the screen), and ScenarioBuilder will not “wait” for a particular screen to load prior to executing mouse clicks. Generally, it is best practice to use anchor images when prompted.

When an anchor image moves, becomes obstructed, or no longer renders on the screen Scenario Recorder will prompt you to update the anchor image;



Just as a real user reacts to a new screen, so Scenario Recorder notices when screens change. When recording, you will be alerted each time the current anchor image becomes invalid for one of the above mentioned reasons. Proceed in one of the following ways:



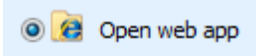
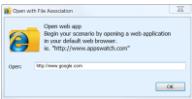

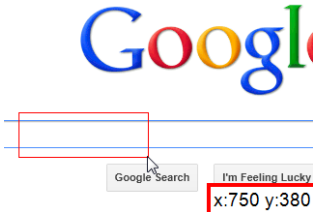
**Continue:** if your anchor image is still valid choose “Continue.” For example, if a pop-up window obstructs the image during recording, and the pop-up will not be a factor during playback, select “Continue.”

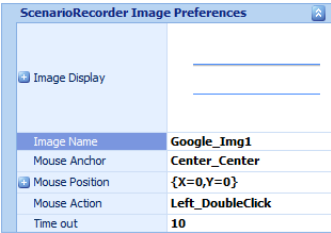
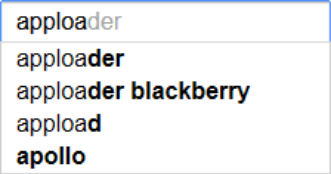
**Capture New Image:** choose this option to update the anchor image. Subsequent mouse coordinates will be relative to the updated anchor image.


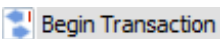
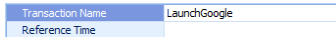

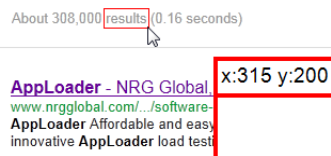

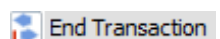
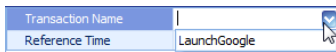
**Cancel:** choose only when you want to continue without using an anchor image. Subsequent mouse coordinates will be absolute and mouse actions will play back without regard for images rendered on the screen. The “Cancel” option suspends further anchor image verification until the next image is captured during the Scenario Recorder session.


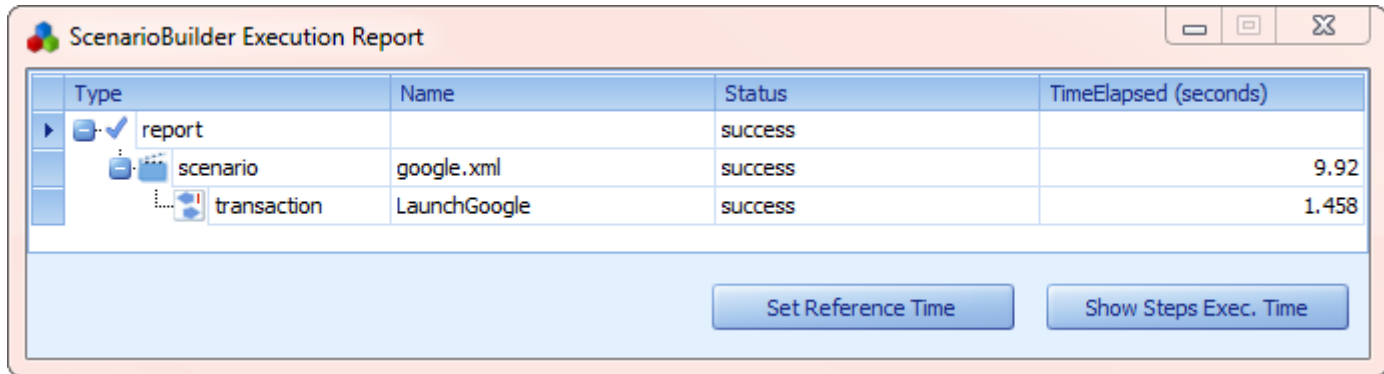


## CREATING YOUR FIRST SCENARIO WITH SCENARIO RECORDER

The following tutorial will help you become familiar with some common Actions used in Scenario Recorder, as well as demonstrate its ease and functionality. This step-by-step walk-through starts with the launching of ScenarioBuilder, takes you through building a scenario that opens Google and conducts a search, and concludes with the playback of the Scenario:

<u>Description</u>	<u>Screenshot</u>	<u>Procedure</u>
Create a New Scenario		Launch ScenarioBuilder and click the “New Scenario” link on the “Splash Screen”.
		Note: if ScenarioBuilder is already open, click the “New Scenario” icon in the tool bar.
Launch Google in your default web browser;		Select “Open Web App” in “New Scenario Properties” form and click “OK”.
		In the “Open” field, type <code>http://www.google.com</code> . Press <Enter>.  <b>Result:</b> Google home page opens in default web browser.
Ensure Google website has loaded before entering search criteria (also referred to as “keeping scenario synchronized”);		Click “Capture Image” icon from the Scenario Recorder menu.
		Capture the middle section of the search box from the Google home page. Avoid the blinking cursor.

		<p>Keep the default “Image Preferences” settings. Press &lt;Enter&gt;.</p> <p><u>Result:</u> scenario looks for and clicks inside the Google search box.</p>
<p>Enter search criteria “AppLoader” into Google search box;</p>		<p>Type <i>AppLoader</i>. Press &lt;Enter&gt;.</p>

Measure the time it takes for Google to return results;		Click “Actions” icon on Scenario Recorder menu.
		Click “Begin Transaction” from “Actions” pop-up menu.
		In the “Transaction Name” property field on “Quick Add” form, type <i>LaunchGoogle</i> . Press <Enter>.
		Click “Capture Image” icon on the Scenario Recorder menu.
		Capture the word “results” on the Google page ( <i>Google will always return the word “results”, regardless of the search criteria</i> ).
		Change the “Mouse Action” in the “Image Preferences” settings to “None”. Press <Enter>.
		Click “Actions” icon on Scenario Recorder menu.
		Click “End Transaction” from “Actions” pop-up menu.
		In the “Transaction Name” property field on “Quick Add” form, select <i>LaunchGoogle</i> from drop-down box. Press <Enter>.
		<u>Result</u> : “LaunchGoogle” Transaction measures search response time.

Play Scenario;		<p>Click “Play Scenario” icon on Scenario Recorder menu.</p> <p><u>Result:</u> scenario plays from beginning. Upon completion, ScenarioBuilder’s “Execution Report” pops-up.</p>
		
Set Reference Time;		<p>Click “Set Reference Time” button to establish a performance standard and close Execution Report.</p>
Stop recording;		<p>Click “Stop Recording” icon on Scenario Recorder menu.</p> <p><u>Result:</u> Scenario Recorder closes and ScenarioBuilder displays.</p>

Your complete scenario appears in the Scenario Window;

Scenario		
Name	Content	Step
scenario	google.xml	
Open with File Association	http://www.google.com	1
Click On Image	google1.bmp	2
Mouse Move	Use Anchor, 0, 0	3
Mouse Click	Left, DoubleClick	4
Type Text	aploader	5
Function Keys	Press, Enter	6
Begin Transaction	LaunchGoogle	7
Find Image	google2.bmp	8
End Transaction	LaunchGoogle	9

## SCENARIO DESIGN BASICS

ScenarioBuilder provides a broad scope of end-user Actions to facilitate the creation of virtually any scenario. We've just scratched the surface in the preceding tutorial. For a complete glossary and explanation of all the ScenarioBuilder Actions, go to [Index A](#) at the back of this guide.

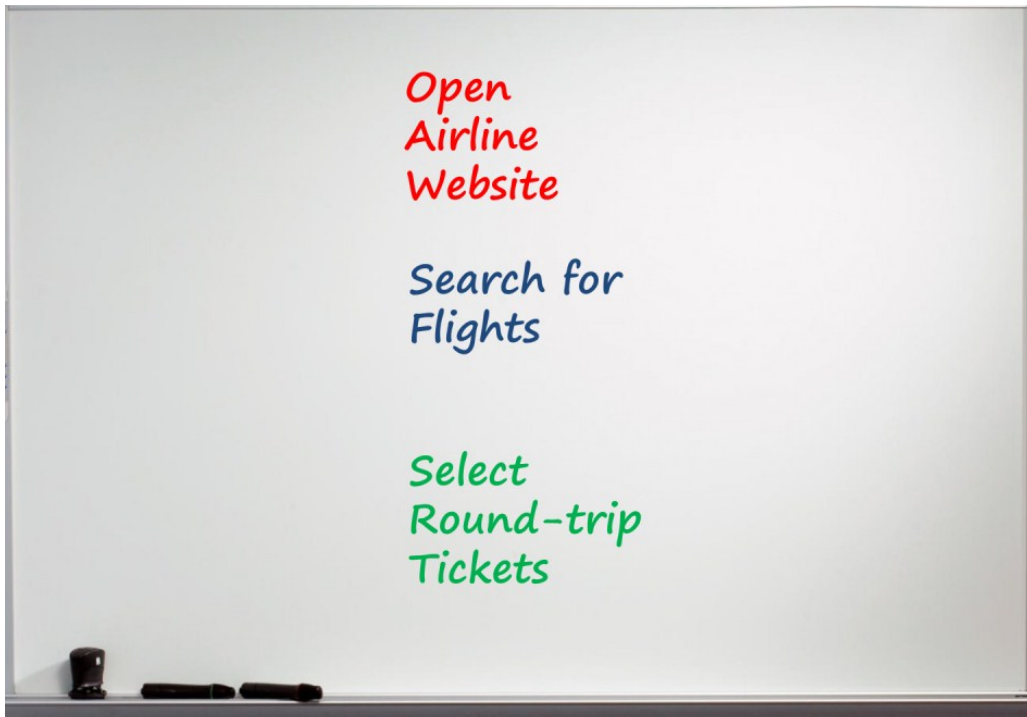
Note that not all Actions operate on the application under test. Some Actions operate on the scenario itself. For example, as demonstrated in the preceding tutorial, a "Transaction" directs ScenarioBuilder to measure the steps contained between its start and end points. And a Component, while technically an Action, is really more of a structural element in the scenario.

Let's take a close look at Components and their role in scenario designing. It's tempting to jump right in and begin building scenarios (in fact, we encourage you to do so, as it's a great way to learn about ScenarioBuilder). However, before embarking on actual test case scenarios, we encourage you to step back and take a breath. The best practice is to start with a diagram outlining the business processes you want to test. The flow chart that you sketch will most likely become the blueprint for your scenario.

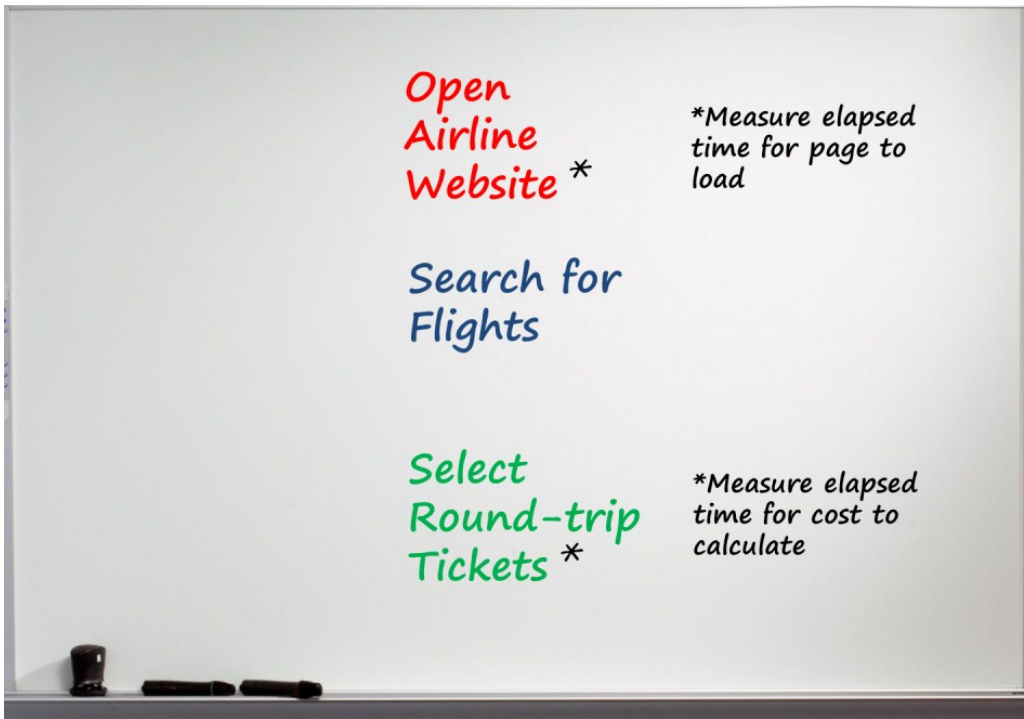
---

### DESIGN OUTLINE 1 – AIRLINE TICKET WEBSITE

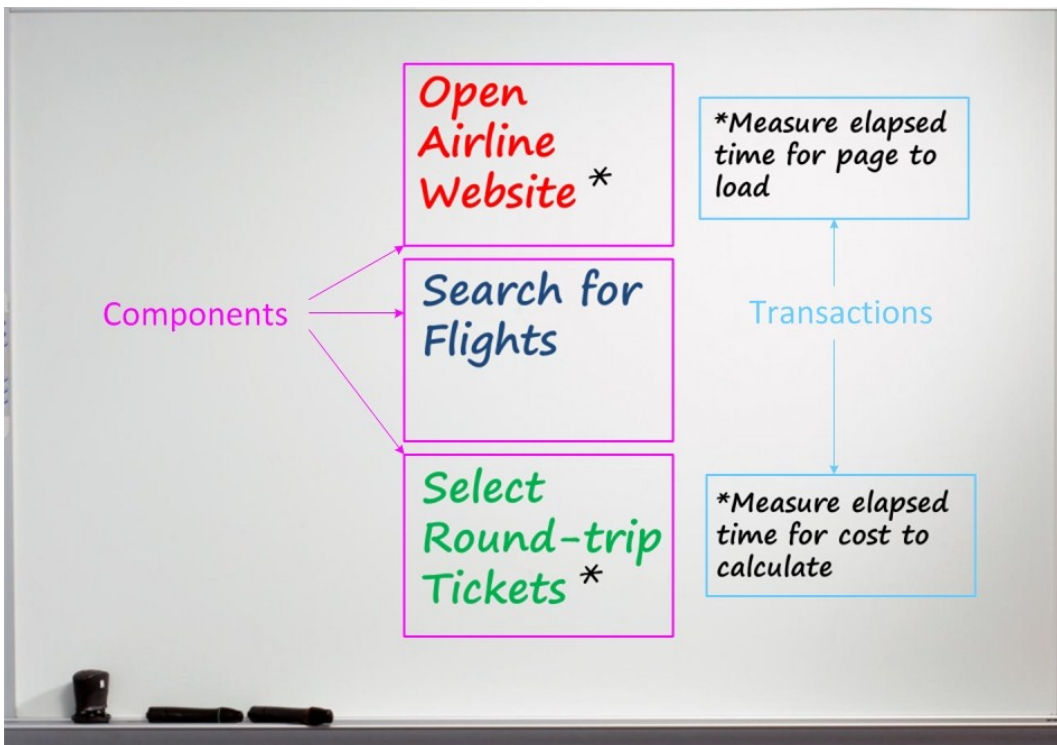
To create a scenario testing an airline's website, for example, you might break out three general processes – *Open Airline Website*; *Search for Flights*; *Select Tickets*;



Now, say you want to know not only how long the processes take overall, but how long specific functions within the processes take. For example, the time the website takes to load, as well as the time it takes to calculate a "total trip cost" for the flights specified;



This simple overview has provided a solid template for a scenario. Below, see how Components and Transactions can be ascertained from the chart;



## DESIGN OUTLINE 2 – SEARCH FOR AN ACCOUNT IN CRM

Here's another example, this one defines a test of a CRM application. The test breaks down into three segments – *Login to CRM application; Look up an account; Logoff of application*. Again, these segments establish the natural structure for the scenario. Notice the outline also includes the Actions (in plain terms) to be executed under each Component, further taking the guesswork out of scenario building:

<p><b>Component 1: Login to CRM</b></p> <p>Actions include:</p> <ul style="list-style-type: none"> <li>• Launch CRM App</li> <li>• Enter login info</li> <li>• Press enter</li> <li>• Wait for CRM home screen</li> </ul>	
<p><b>Component 2: Lookup An Account</b></p> <p>Actions include:</p> <ul style="list-style-type: none"> <li>• Click search icon</li> <li>• Enter account info</li> <li>• Press enter</li> </ul>	<p>Use <b>Execute Component</b> to call a Component into your scenario and use <b>Define Component</b> to add Actions to the Component.</p>
<p><b>Component 3: Logoff CRM</b></p> <p>Actions include:</p> <ul style="list-style-type: none"> <li>• Click logoff</li> <li>• Confirm logoff</li> <li>• Wait for logoff screen</li> </ul>	

Take it one step further and add Transaction markers. If you want to know specifically how long it takes for the application to return the search results when a user looks up an account, insert a Transaction “Begin” and “End” Action in Component 2:

<div data-bbox="191 396 678 808" style="background-color: #90EE90; padding: 10px; border: 1px solid black;"> <p><b>Component 2: Lookup An Account</b></p> <p>Actions include:</p> <ul style="list-style-type: none"> <li>Click search icon</li> <li>Enter account info</li> <li><b>Begin Transaction</b></li> <li>Press enter</li> <li>Wait for CRM home screen</li> <li><b>End Transaction</b></li> </ul> </div>	<p><i>Use <b>Begin Transaction</b> to establish the starting point of your measurement, and <b>End Transaction</b> to mark the stopping point. Provide a meaningful name for Transactions as that name will be used in graphs and reports.</i></p>
---	--

### DESIGN OUTLINE 3 – ENTER HOURS IN A TIMESHEET APPLICATION

This last example shows how to organize your scenario into a basic chart format using a spreadsheet. This example outlines a scenario for testing a Citrix-based timesheet application. There are three main tasks involved:

- Load Citrix and start the application
- Enter hours
- Close the application

Hierarchy	Scenario/Tasks	Type	Description	What to do on Failure?
1	Timesheet	Scenario	User enters timesheet data and submits for approval	Close TS application
2	Start TS	Component	Load Citrix and logon to the application	
3	Logon	Transaction	Click on signal button and wait until the first form displays	
2	Enter hours	Component	Select everyday of the week and enter the hours worked	Delete entered hours
3	Submit TS	Transaction	Click on submit TS button on wait for successful message	If resource busy, wait 2 seconds then return to submit again
2	Close TS	Component	Logoff and close TS	
3	Logoff	Transaction	Click on logoff link and wait for sign on screen to appear	

Each of the three main tasks represents a Component in the scenario. In addition, the chart marks Transactions to measure the time it will take to perform key tasks within the Scenario:



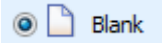
- How long does it take to log on to the application?
- How long does it take to submit the hours? (i.e. receive a successful confirmation)
- How long does it take to log off from the application?

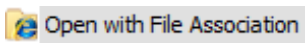
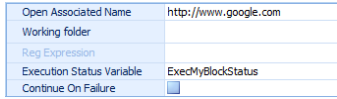

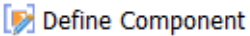
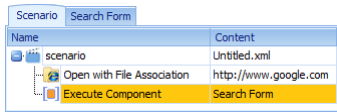

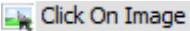
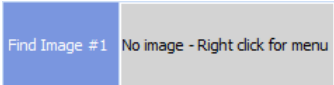
Download the following sample chart in the support area at [www.nrgglobal.com](http://www.nrgglobal.com) to help you lay out your Scenario:

Hierarchy	Scenario/Tasks	Type	Description	What to do on Failure?
1	Scenario Name	Scenario	High level description of what the scenario is trying to achieve	What should be done when the scenario fails
2	Component1 name	Component	Describe what the component does	What should be done when Component1 fails
3	Trx1 name	Transaction	Describe the first transaction	N.A.
2	Component2 name	Component	Description of Component2	What should be done when Component2 fails
3	Trx2 name	Transaction	Description of Transaction2	What should be done if transaction fails (i.e. timeout)
2	Component 3 name	Component	Description of Component3	N,A
3	Transaction name	Transaction	Description of Transaction2	N.A

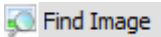
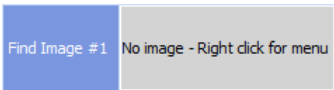
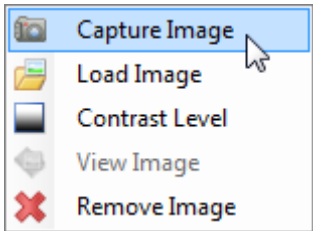
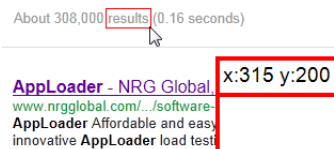
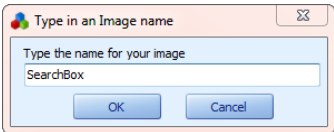

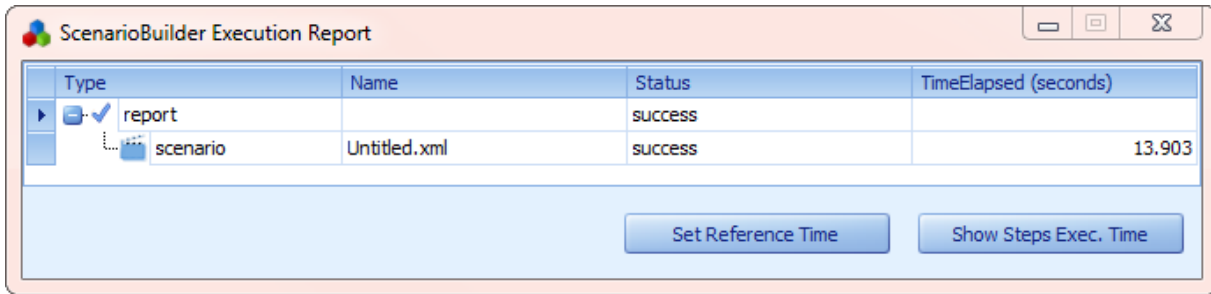
## CREATING YOUR FIRST SCENARIO WITH SCENARIOBUILDER

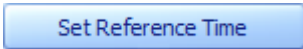
Now, we'll re-create the "Google Search" Scenario demonstrated in the preceding tutorial, but this time using the ScenarioBuilder interface. Also, we'll utilize a Component in this version and omit the Transaction;

<u>Description</u>	<u>Screenshot</u>	<u>Procedure</u>
Create a New Scenario		Launch ScenarioBuilder and click the "New Scenario" link on the "Splash Screen".
		Note: if ScenarioBuilder is already open, click the "New Scenario" icon in the tool bar.
Open ScenarioBuilder in default view;		Select "Blank" in "New Scenario Properties" form and click "Okay".




Open Google website with default browser;		Double-click “Open with File Association” from Actions pane (under “Applications”).
		In the “Open Associated Name” field in “Properties” pane, type <i>http://www.google.com</i> .
		Click “Play Selected Step” icon in tool bar.  <b>Result:</b> Google home page opens in default web browser.
Define Component;		Double-click “Define Component” from Actions pane. Name Component “Search Form”.
		<b>Result:</b> ScenarioBuilder adds a “Search Form” tab and an “Execute Component” Action to Scenario Window.
<b>Important:</b> Switch Scenario Window focus to “Search Form” Component;		Click “Search Form” tab to display Component in Scenario Window.
Ensure Google website has loaded before entering search criteria (also referred to as “keeping scenario synchronized”);		Double-click “Click On Image” from the Actions pane (under “Images”).
		Right-click “No image – Right click for menu” in “Properties” pane.

Continued...		Click "Capture Image" from pop-up menu.
		Press <Ctrl> and capture the middle section of the search box from the Google home page. Avoid the blinking cursor.
		Enter <i>SearchBox</i> as the name for the image. Click "OK".
		Click "Play Selected Step" icon in tool bar.
Enter search criteria "AppLoader" into Google search box;		Double-click "Type Text" from the Actions pane (under "Keyboard").
		In "Text" field in "Properties" pane type <i>AppLoader</i> . Click "OK".
		Click "Play Selected Step" icon in tool bar.
Submit search;		Double-click "Function Keys" from the Actions pane (under "Keyboard"). Keep default Action "Property <Enter>".








Verify that search results were returned;		Double-click “Find Image” from the Actions pane (under “Images”).
		Right-click “No image – Right click for menu” in “Properties” pane.
		Click “Capture Image” from pop-up menu.
		Press <Ctrl> and capture the word “results” on the Google page ( <i>Google will always return the word “results”, regardless of the search criteria</i> ).
		Enter <i>Results</i> as the name for the image. Click “OK”.
Play Scenario;		Click “Play Scenario” icon on Tool Bar.  <u>Result</u> : Scenario plays from beginning. Upon completion, ScenarioBuilder’s “Execution Report” pops-up.
		

Set Reference Time;		Click “Set Reference Time” button to establish a performance standard and close Execution Report.
---------------------	---	---

Completed scenario appears in the Scenario Window;

Scenario		
Search Form		
Name	Content	Step
 scenario	google.xml	
 Open with File Asso...	http://www.google.com	1
 Execute Component	Search Form	2

Notice there are only two steps in the main “Scenario” tab. The second step calls for the execution of the “Search Form” Component. Click the “Search Form” tab to see the steps held inside the Component;

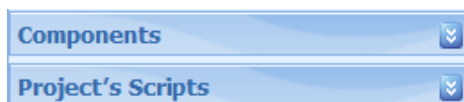
Scenario		
Search Form		
Name	Content	Step
 Define Component	Search Form	3
 Click On Image	SearchBox.bmp	4
 Mouse Move	Use Anchor, 0, 0	5
 Mouse Click	Left, Click	6
 Type Text	uploader	7
 Function Keys	Press, Enter	8
 Find Image	results.bmp	9

## COMPONENTS

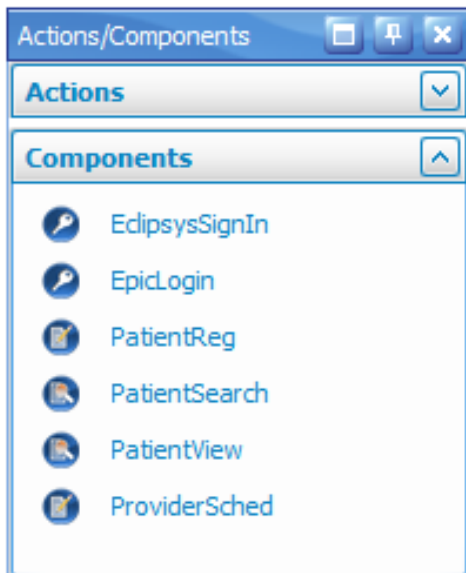
A Component is a user-defined shell within a scenario that contains a single Action or a series of Actions. Think of Components as containers for Actions related to a specific task in the overall process. Breaking up scenarios into Components makes for easier scenario maintenance, and can be helpful in conceptualizing a scenario design.

Components are self-contained units that can be called from any scenario within a Project. Additionally, Components can be exported and imported to and from Projects. If a modification is made to a Component that is being used by multiple scenarios, the Component is updated for all scenarios.

By default, the Components pane is collapsed on the left of ScenarioBuilder inside the Actions/Components window;




Click the down arrow  icon to expand the pane;

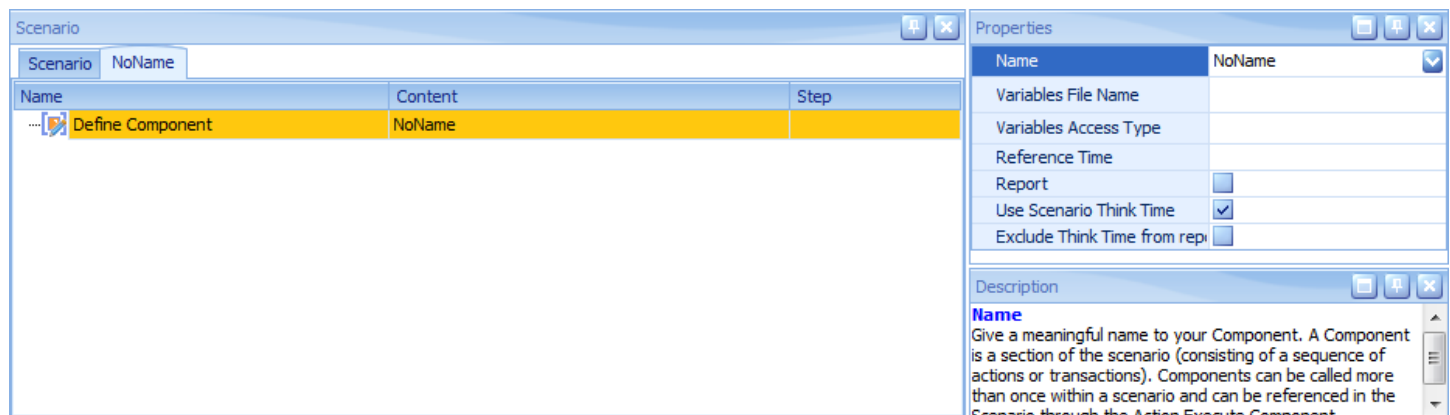


The pane contains the Components in the active Project, which you've created or imported. These Components can be shared among different scenarios throughout the Project, and exported for use in other Projects. All scenarios in the active Project that share a Component in the active Project will inherit any changes made to that Component.

Double-click a Component to add it to your scenario.

## DEFINE COMPONENTS

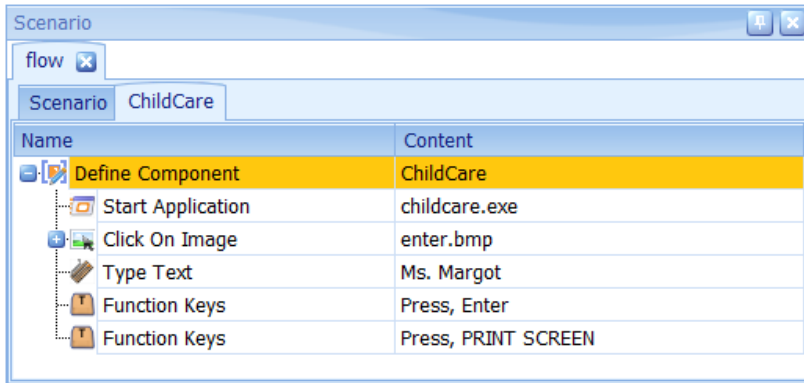
To create a new Component, double-click or drag the "Define Component"  **Define Component** Action from the "Applications" section of the Actions pane into the Scenario Window. ScenarioBuilder adds a tab called "NoName" to the Scenario Window;



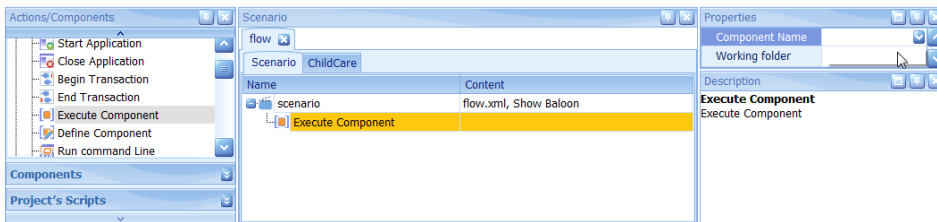
Update the Component's "Name" property with a meaningful name (notice the tab name changes);




With the Component tab selected, add Actions by double-clicking or dragging into the Scenario Window;

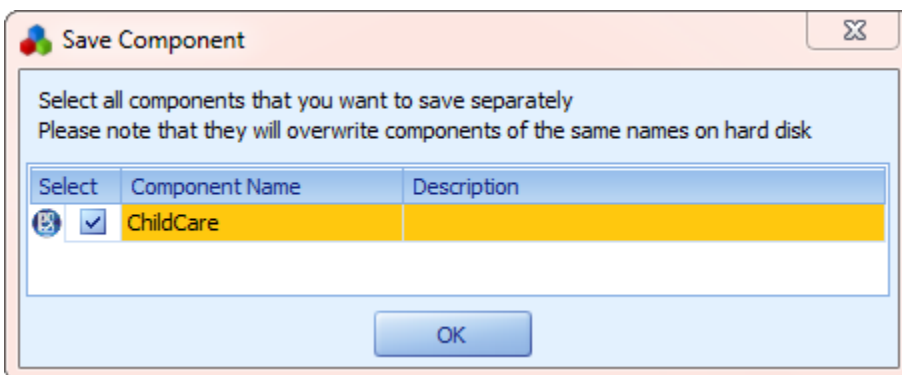


Click the “Scenario” tab. Add an “Execute Component” Action to the scenario. Select the Component from the “Component Name” drop-down box;



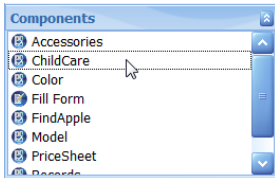
## SAVE COMPONENTS

ScenarioBuilder will prompt you to save Components that have been added or modified as part of the Scenario saving process. Click the “Save Scenario”  toolbar icon, or select “Save Scenario” from the “File” menu. If a new Component has been added to the Scenario, or a pre-existing Component has been modified, the “Save Component” window will open;



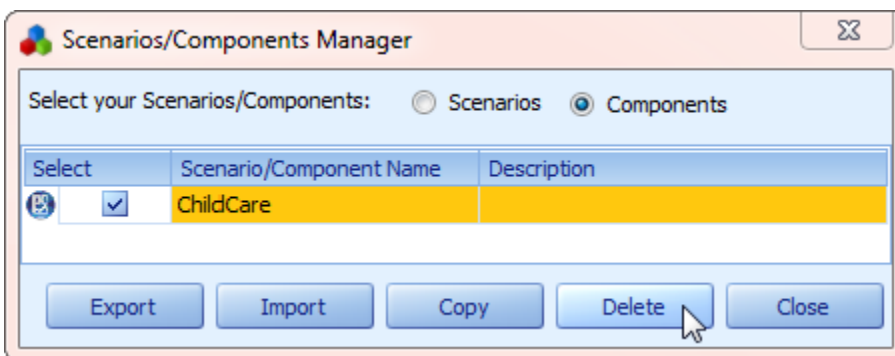
Select the Components you wish to save and click “OK”. Note that when you save a Component that is shared by other Scenarios in a Project, you are updating it globally, meaning any changes are effective for all Scenarios in which it is shared. If this is not your intention, close the “Save Component” window, re-name the Component, and repeat the “Save” process.

After saving the Scenario, the Components pane is updated with any new additions;



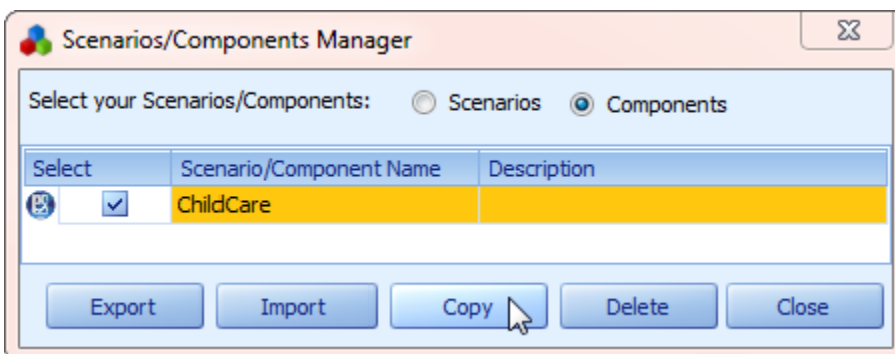
Components will remain in the Components pane for the current Project, unless deleted.

To delete a Component, select “Scenarios and Components Manager” **Scenarios and Components Manager** from the File menu. In the “Scenarios/Components Manager” window click the “Components” radio button. Select the Component to delete and click the “Delete” button;

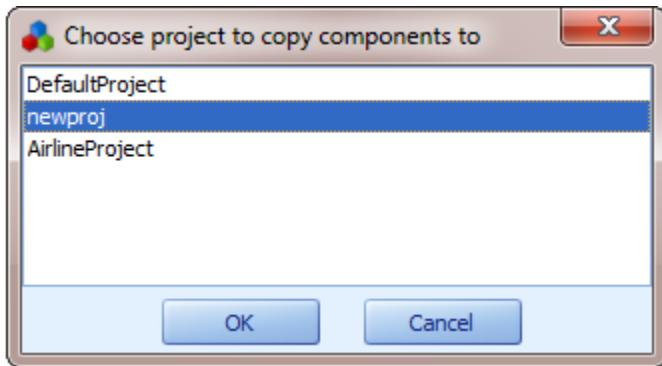


## COPY A COMPONENT TO ANOTHER PROJECT

You can directly copy a Component from the current Project to another Project (provided the Projects reside in the same “Base folder for All Projects” which is established in the “Options” menu). Select “Scenarios and Components Manager” from the File menu. In the “Scenarios/Components Manager” window click the “Components” radio button. Select the Component to copy and click the “Copy” button;



The “Choose project to copy Components to” window pops up. All of the Projects in the current base folder are displayed;



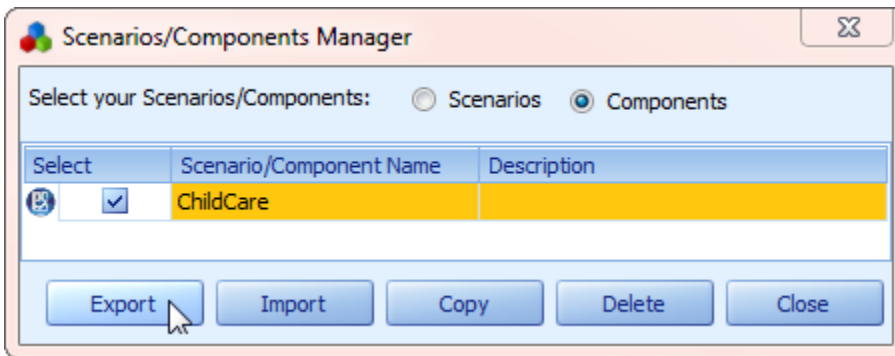
Select the Project to which you would like to copy the Component and click “OK”.

## EXPORT AND IMPORT COMPONENTS

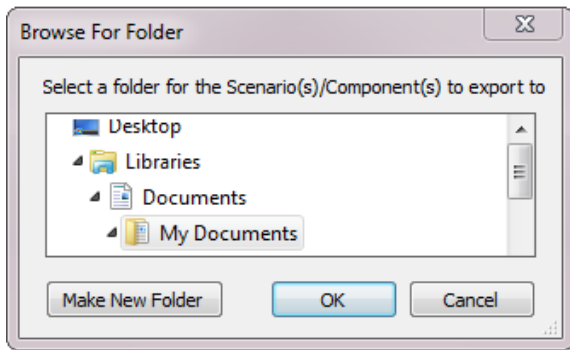
Use “Import” and “Export” to move Components in and out of Projects residing in different work spaces.

### EXPORT COMPONENT

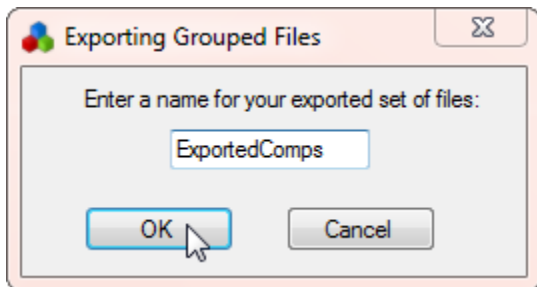
In the “Scenarios/Components Manager” window click the “Components” radio button. Select the Component(s) to export and click the “Export” button;



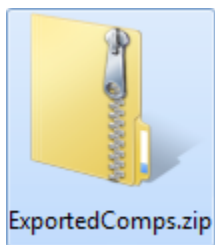
Navigate to an existing folder or create a new folder into which to export. Note that the export folder is a neutral location from which the Component(s) will be imported; it is not intended to be the final location (i.e. the Project folder) for the Component(s). Click “OK”;



The “Exporting Grouped Files” window pops up. Enter a name for the exported set of files and click “OK”;



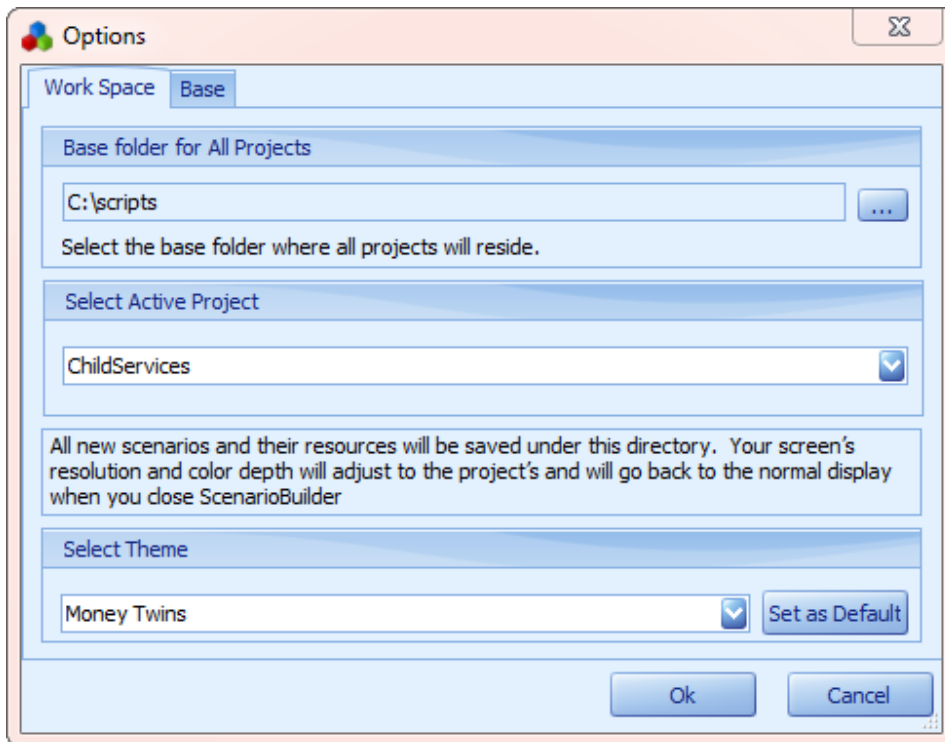
Your exported Component(s) will be saved as a .zip folder in the folder you selected;




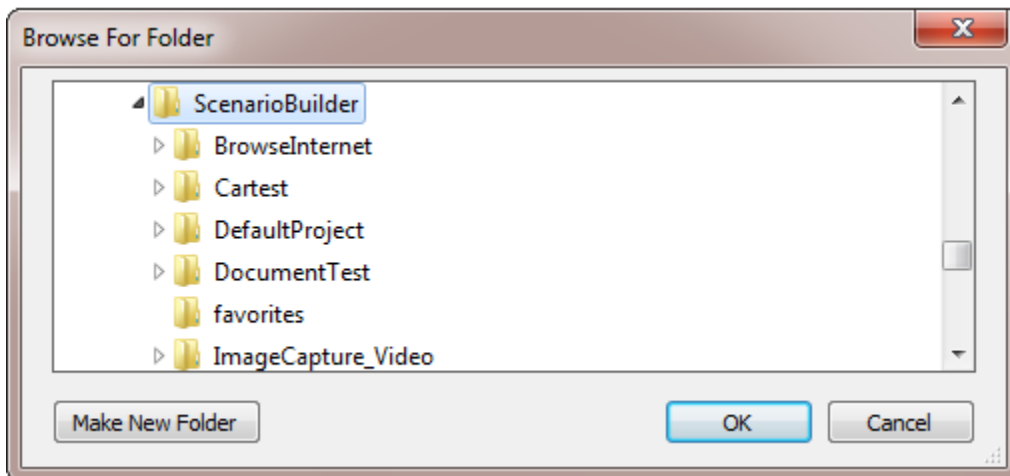

---

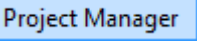
## IMPORT COMPONENT

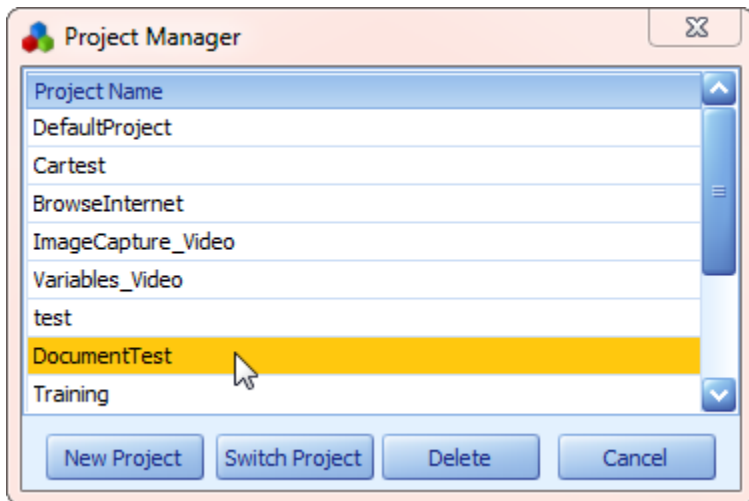
After exporting your Component(s), open the existing Project, or create a new Project into which you wish to import the Component(s). If importing to a Project in a different base folder, open the “Options” menu and click the “Work Space” tab;



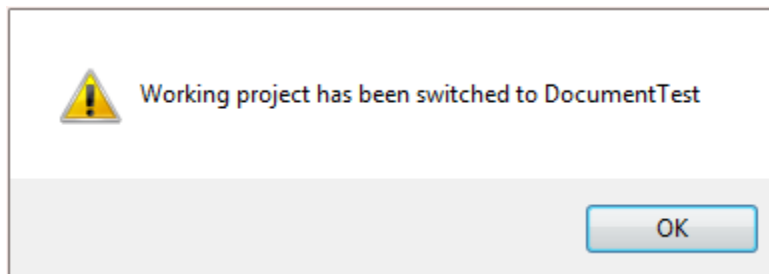
Click the ellipses  button in the “Base folder for All Projects” pane to navigate to the desired Project base folder;



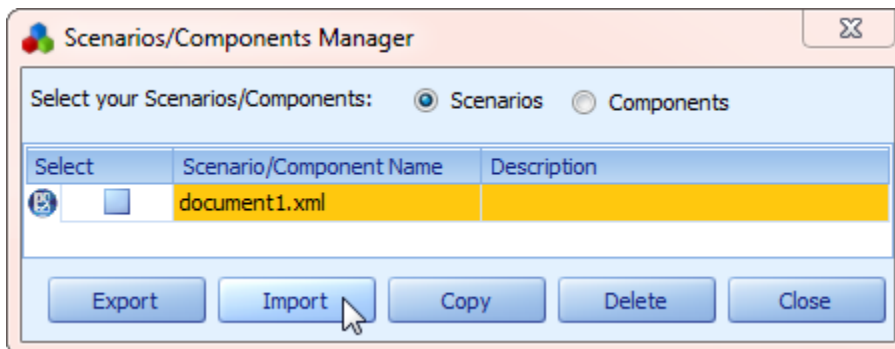
Select the folder and click “OK”. Next, click the “Project Manager”  option in the “File” menu. The Project Manager opens;



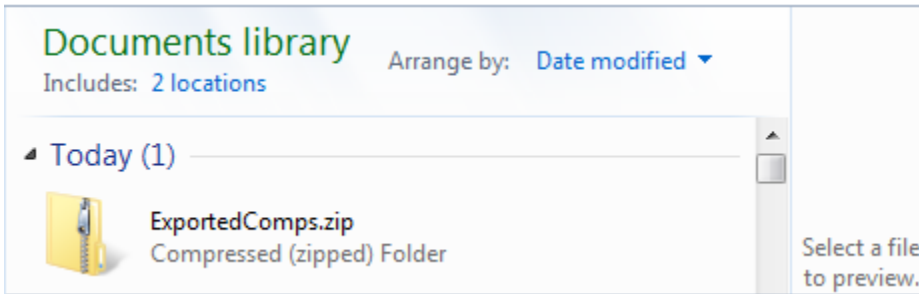
Select the Project to which you want to switch and click “Switch Project” button. ScenarioBuilder will notify you that the Project folder has been switched;



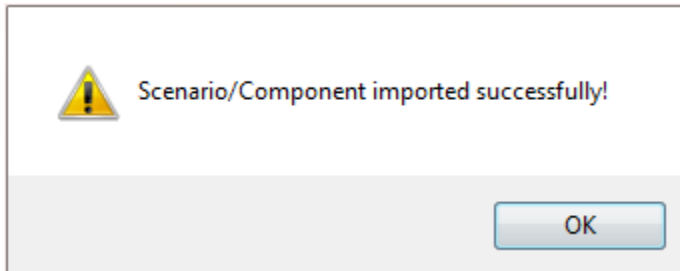
Now that you are working in a new Project folder, you can import the Component(s). Open the “Components and Scenarios Manager” and click “Import” button;



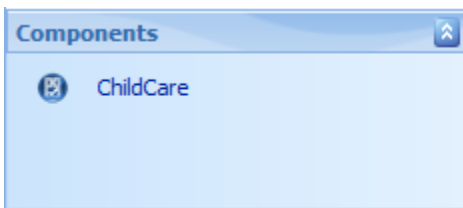
Browse to the folder containing the exported Component(s);



Double-click the desired .zip file. ScenarioBuilder will acknowledge the import;



Imported Component(s) will appear in the Project's "Components" pane;

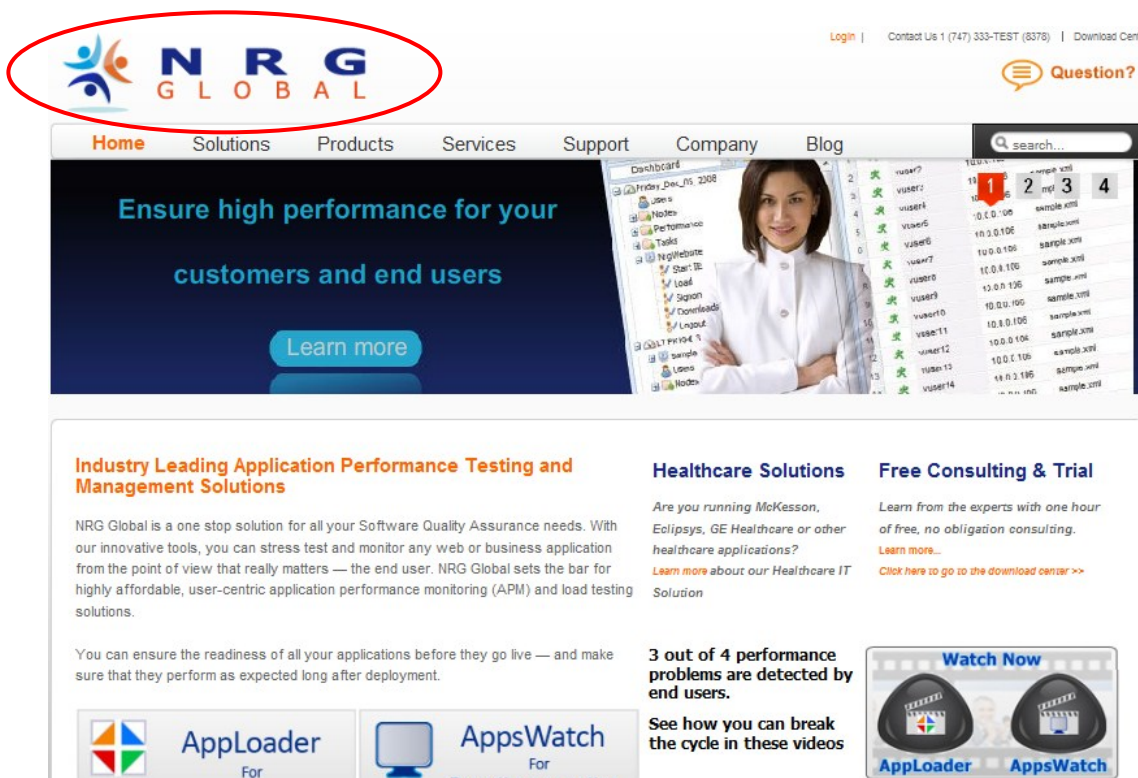


## WORKING WITH IMAGES

Image Actions are ScenarioBuilder's most versatile Actions. Images can be used to start an application, confirm that an application or web page has loaded, or hold until an application has closed.

An image can be any kind of bitmap – a company logo on a client application; a photo on a web page; or a data entry field on an application form.

As an example, if testing the <http://www.nrgglobal.com> website, we could verify that the home page has successfully loaded by asking the scenario to check for the NRG Global logo. A “Find Image” Action would search for the logo until its time out period is exhausted. Subsequent Actions would hold until the outcome of the “Find Image” Action is determined.



We recommend that you visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Images Actions” video tutorial.

## THINK LIKE A REAL USER

When building a Scenario, think in terms of the end user. How will they interact with the application? What will they do? For example, if the application involves a sign-up or data entry form, a user will not start entering data until the field has properly loaded on the screen. In ScenarioBuilder, that synchronization can be achieved with Image Actions.

## IMAGE ACTIONS KEEP SCENARIOS SYNCHRONIZED

Check out the following example which demonstrates a simple scenario that opens a company's online job application form, and enters text into the data fields. First the website has to be opened. ScenarioBuilder uses the "Open with File Association" Action to open the URL;

Scenario	
Name	Content
scenario	employmentApp.xml
Open with File Association	http://www.CompanyEmployment.com

The website opens to the job application form;

Basic Job Search

Search Criteria

Specify your job search criteria, then click "Search for Jobs".

Job Number

Keywords

Job Field

Green Coffee & Tea Mgmt.

All

Add Job Field

Location

All

Add Location

Search for Jobs

Clear

Hide Search Criteria | Save this Search

Search Tips

You can search jobs by selecting relevant criteria in the drop-down menus. You can also use a job number or a keyword.

Saving searches

You can save the current search for reuse by clicking "Save this Search" at the top of the page. Your searches will be saved in the "My Saved Searches" section, under the "My Jobpage" tab.

Advanced job search

To perform a job search using advanced search criteria, click the "Advanced Search" tab and select the relevant criteria.

Applicant Accommodation

Starbucks Coffee Company is committed to offering reasonable accommodations to job applicants with disabilities. If you need assistance or an accommodation due to a disability, please contact us at 206-318-0660 or via email.

RSS

The next step in the scenario demonstrates a fundamental concept in ScenarioBuilder which you'll use over and over. With a "Click on Image" Action, ScenarioBuilder merges real user intelligence with automation. An image of the "Job Field" will serve two purposes;

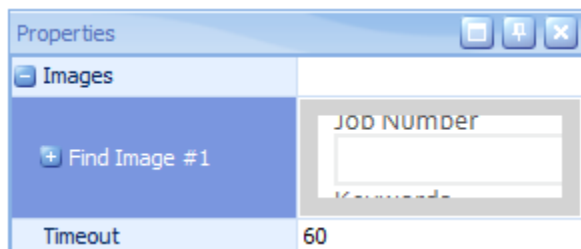
Job Number

Keywords

Job Field

x:153 y:214 w:94 h:40

First, the scenario will look for the field, as a real user would (notice that enough of the field heading is included in the image captured to ensure that it is uniquely identifiable by ScenarioBuilder). ScenarioBuilder will delay subsequent Actions until this image is located. Set the “Time Out” property to provide adequate time for the form to load (factor in a longer response time for multiple users interacting concurrently in your test environment);



Just as a real user would not enter data into a form that hasn't loaded, so ScenarioBuilder waits for the form to load. Next, the “Click on Image” with its “Mouse Click” child Action places the cursor into the data field so that the subsequent Action, in this case “Type Text”, will succeed. A real user knows to click onto a form to get their cursor focused properly before typing. This is how ScenarioBuilder achieves proper mouse focus;

Scenario	
Name	Content
scenario	employmentApp.xml
Open with File Association	http://www.CompanyEmployment.com
Click On Image	JobNumber.bmp
Mouse Move	Use Anchor, 0, 0
Mouse Click	Left, Click
Type Text	70259

## COMBINE IMAGE AND LOOP ACTIONS TO PAGE DOWN UNTIL IMAGE IS FOUND

Sometimes an image you are looking for on a web page may appear “below the fold”. It might require one, two or many “page downs” before it appears. A real user would quickly scan the screen for the desired image, and page down until the image appears. In ScenarioBuilder that same intelligence can be achieved with Image and Loop Actions. The following example opens Amazon.com;

Scenario	
Name	Content
scenario	Amazon.xml
Open with File Association	http://www.amazon.com

Next, the scenario uses an image of the Amazon logo to ensure that the web page has loaded;

Find Image	amazonLogo.bmp	2
------------	----------------	---

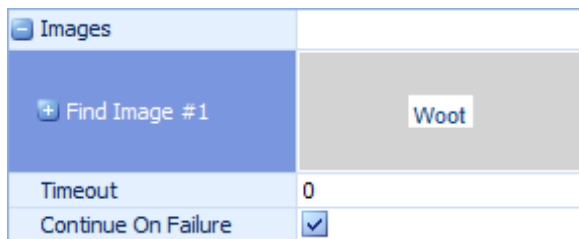
The logo image has a generous “Timeout” value, ensuring that there is ample time for the page to load;



Now the scenario searches for the “Woot” icon located at the bottom of the Amazon home page;



The “Timeout” value for the “Woot” image is minimal;





The scenario knows that the page has already loaded (previous step found the logo image), and within milliseconds knows whether or not the “Woot” image is present. As soon as it scans the screen and realizes the image is not present, it moves on to the next step (note the image’s “Continue on Failure” property is enabled, otherwise the scenario would fail and stop after the first attempt to find the image). The complete scenario looks like this;

Scenario		
Name	Content	Step
scenario	Amazon.xml	
Open with File Association	http://www.amazon.com	1
Find Image	amazonLogo.bmp	2
Loop	5	3
Click On Image	Woot.bmp	4
Mouse Move	Use Anchor, 0, 0	5
Mouse Click	Left, Click	6
Exit	loop	7
Function Keys	Press, Page Down	8

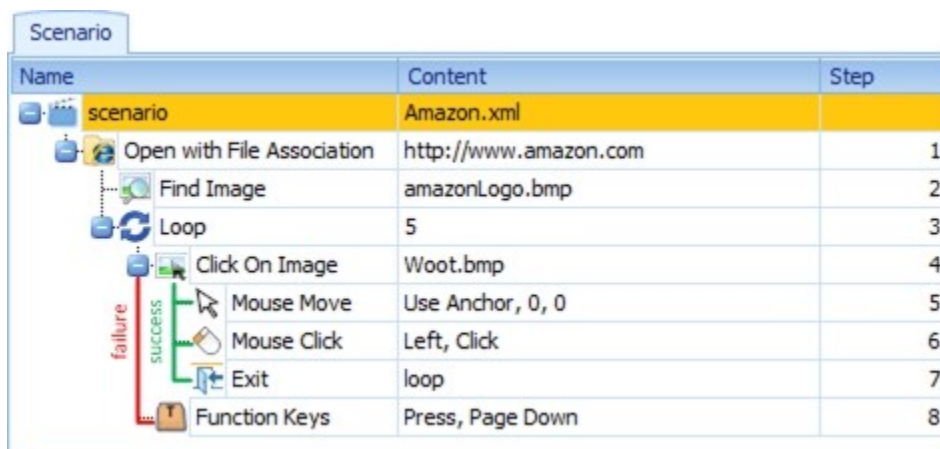
It’s important to note the use of indentation in the example. Indented Actions are “children” of the Actions under which they’re indented. Child Actions only execute if their parent Action has succeeded. The

relationship of the steps in a loop (parent to child) must be set up properly or the loop won't function as expected.

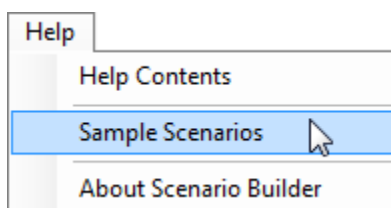
Step 4 is the first step inside the loop. It's considered inside the loop because it is indented, and therefore a child of the Loop Action.

Use the "Move Right"  and "Move Left"  tool bar icons to shift steps from parent to child or vice versa.

If step 4 succeeds, it will advance through its child Actions - steps 5, 6 and 7 - culminating in an *exit* from the loop. If step 4 fails, it will advance (only because "Continue on Failure" is enabled), bypassing its child Actions, to step 8. Step 8 executes a "Page Down" function and returns the scenario to step 3 for the next iteration through the loop;



For more about Loops, please check out "Conditional Statements" in our ScenarioBuilder Training Videos series at <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos>. Also, see the "Loop" example in the "Sample Scenarios" section of ScenarioBuilder's "Help" menu;



## WAIT FOR IMAGE TO DISAPPEAR BEFORE ADVANCING

The "Wait for Image to Disappear" property on an Image Action can be used to keep a scenario synchronized. Say your web application does not always respond when icons are clicked. Real users know to re-click and wait for the page to load before advancing to the next step, but how do automated users know how to react?

The following scenario opens Yahoo.com and then uses a "Click on Image" Action to click on the link to the "Autos" page;

Scenario	
Name	Content
scenario	Untitled.xml
Open with File Association	http://www.yahoo.com
Click On Image	Autos.bmp

Next the scenario uses a “Find Image” Action with the same image of the “Autos” link, but with the “Wait for Image to Disappear” property enabled;

Wait for Image to Disappear ☒

Notice the Action changes to the inverse “Wait Image Disappear” Action;

Scenario	
Name	Content
scenario	Untitled.xml
Open with File Association	http://www.yahoo.com
Click On Image	Autos.bmp
Mouse Move	Use Anchor, 0, 0
Mouse Click	Left, Click
Wait Image Disappear	Autos.bmp

The last piece is to add an “On Failure” section which kicks in only if the image does *not* disappear. Two Actions are in the “On Failure” section, a “Click on Image” Action to click that same “Autos” link followed by a “Resume” Action which instructs ScenarioBuilder to go back to the point of the failure and try again. The “Resume” has a “Retries” property which determines the number of times to loop through the “On Failure” section before giving up. In this example it’s set to 3 times.

The complete scenario looks like this;

Scenario		
Name	Content	Step
scenario	Untitled.xml	
Open with File Association	http://www.yahoo.com	1
Click On Image	autos.bmp	2
Mouse Move	Use Anchor, 0, 0	3
Mouse Click	Left, Click	4
Wait Image Disappear	Autos.bmp	5
On Failure		6
Click On Image	Autos.bmp	7
Mouse Move	Use Anchor, 0, 0	8
Mouse Click	Left, Click	9
Resume	3	10

## ALLOW FOR IMAGE VARIATIONS

A real user is not thrown off by a slight variation in an icon or the color of a button. However, when automating test scenarios it's important to take into account possible variations and make allowances for them.

ScenarioBuilder provides various means to ensure that images are found under the widest variety of circumstances. For example, a hyperlink will often change colors once it has been followed during a browser session. This can be handled effectively by first capturing the image in its “un-followed” state;

[Load testing tools | Application Testing Software | NrgGlobal](#)

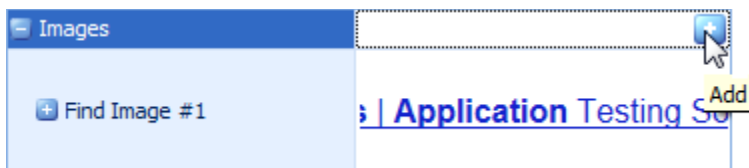
[www.nrgglobal.com/Healthcare-Testing](http://www.nrgglobal.com/Healthcare-Testing)

Load testing tools | Application Testing Software | NrgGlobal ... Testing Healthcare

Applications: How to Set up a Load Test on a Healthcare Application ...

x:292 y:414 w:462 h:23

Once captured, go to the Image Action's properties and click the “+” in the “Images row;



Notice a new property field called “Find Image #2” is added to the pane. Right click on the “No image-Right click for menu” field and capture another image of the hyperlink, this time in its “followed” state;

[Load testing tools | Application Testing Software | NrgGlobal](#)

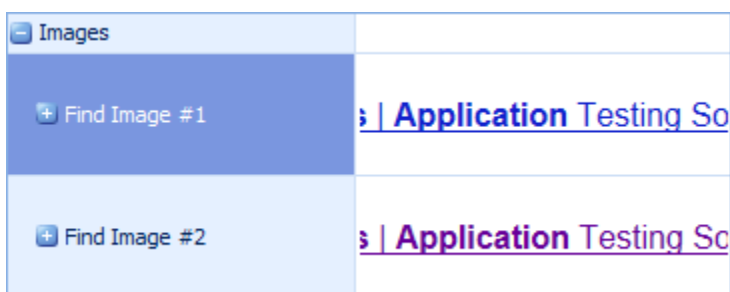
[www.nrgglobal.com/Healthcare-Testing](http://www.nrgglobal.com/Healthcare-Testing)

Load testing tools | Application Testing Software | NrgGlobal ... Testing Healthcare

Applications: How to Set up a Load Test on a Healthcare Application ...

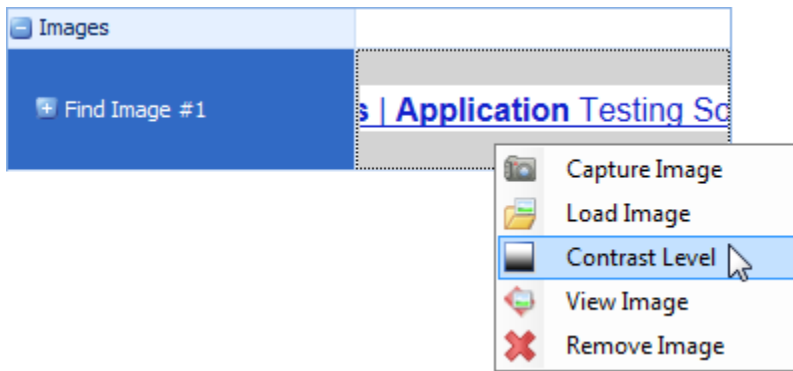
x:294 y:500 w:459 h:21

Now the properties pane includes two images for the Action to seek;

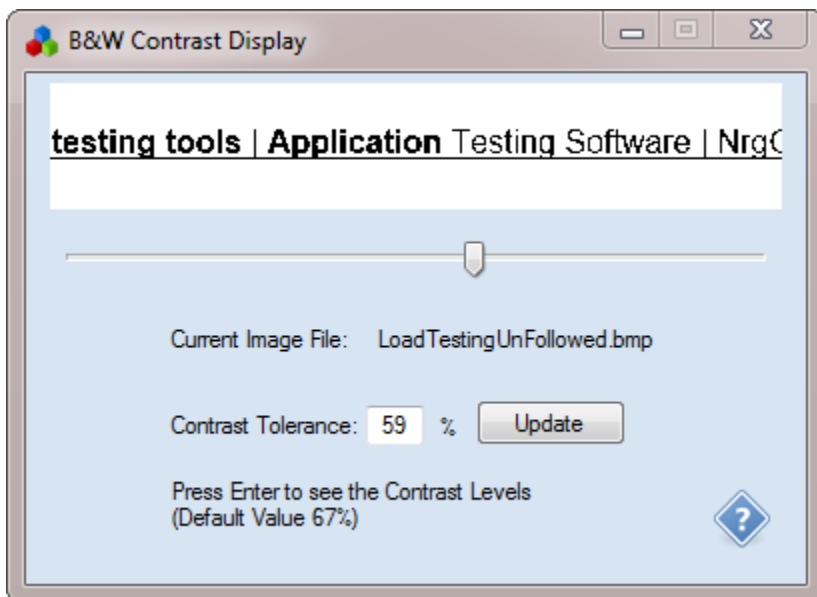


Finding *either* image will make the Action a success.

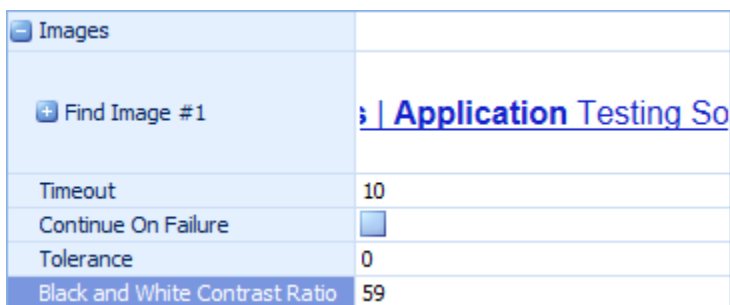
An alternate method to handle the problem of a color-changing hyperlink is to search for the image in black and white. ScenarioBuilder provides a tool to set a black and white contrast ratio for an image, which factors out color and searches for the image in grey scale. Right click the “No image-Right click for menu” field in the properties window and select “Contrast Level”;



The “B&W Contrast Display” tool opens. Use the sliding scale to establish a black and white “Contrast Tolerance” percentage for the image;



When you settle on a “Contrast Tolerance” that displays the image accurately without under or over saturation, click “Update” to send the setting to the Image properties;



If the “Black and White Contrast Ratio” property is set to any value greater than zero, ScenarioBuilder will look for the image in black and white. To return the Image Action to color, simply reset the value to zero.

For subtle irregularities in image renderings, ScenarioBuilder provides a “Tolerance” property which will increase the image matching error margin;

Tolerance	15
-----------	----

Use this property to establish the maximum percentage of differences allowed between the stored image and the desktop rendering (the higher the percentage, the greater the margin for error). Be aware that a high tolerance (certainly anything over 50%) can lead to false positives.

## IMAGE APPEARS TWICE ON THE DESKTOP

Sometimes a form will have two identically labeled fields. The real user knows which one to click into, but how does ScenarioBuilder know? With the “Search Area” property you can instruct the scenario to search for an image in a specific area of the screen. The following form has duplicate “address” fields with identical field labels;

### Enter your old address.



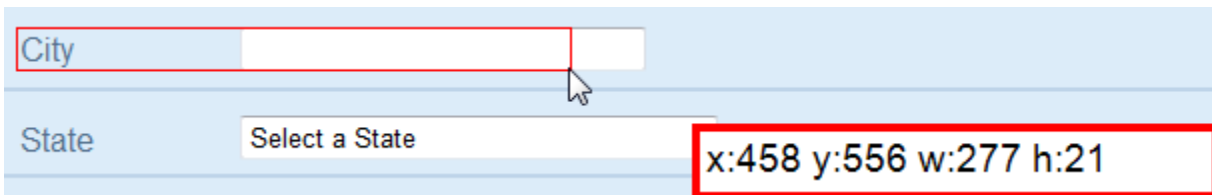
Street	<input type="text"/>
	<small>Include your apartment, suite number, OR PO Box, if applicable</small>
City	<input type="text"/>
State	<select><option>Select a State</option></select>
ZIP Code	<input type="text"/> <a href="#">Auto-fill your ZIP Code</a>

### Enter your new address.

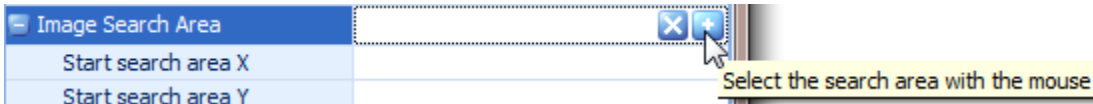


Street	<input type="text"/>
	<small>Include your apartment, suite number, OR PO Box, if applicable</small>
City	<input type="text"/>
State	<select><option>Select a State</option></select>
ZIP Code	<input type="text"/> <a href="#">Auto-fill your ZIP Code</a>
<input type="checkbox"/> This residence was built in the last 6 months. <a href="#">More Info</a>	

First, with a “Click on Image” Action capture an image of the label and field;

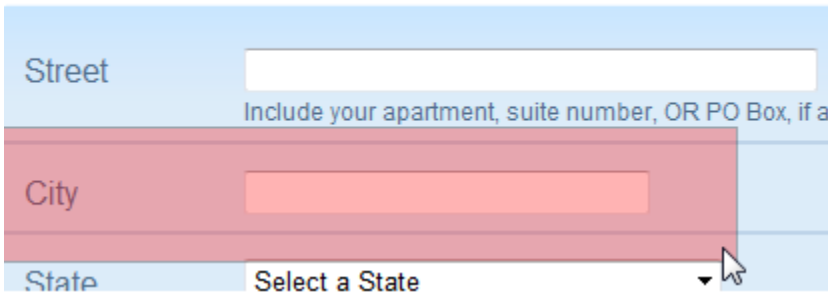


Click the “+” icon in the Image Action’s “Image Search Area” property field to activate the “Search Area” tool;



Use the tool to highlight the area of the desktop where you want the search to take place;

Enter your old address.



ScenarioBuilder will enter the absolute coordinates of the search area into the Image Action properties and select the “City” field based on these coordinates;

Image Search Area	
Start search area X	442
Start search area Y	530
End search area X	918
End search area Y	599

Please click here for more about [Image Actions & Properties](#). Also, visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Images Actions” video tutorial.

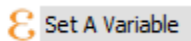
## WORKING WITH SCENARIO PARAMETERS

With Scenario Parameters, data input can be varied to emulate real world application usage. Scenario Parameters can be:

- Pre-defined values that follow specific formats, such as a date, time, or number.
- *Execution Status Variables* determined by the success or failure of an Action.
- User-defined sets of values contained in CSV files, such as names, addresses, birth dates, etc., which can be accessed sequentially or randomly by your scenario.

### ADDING PRE-DEFINED VARIABLES TO A SCENARIO

The “Set A Variable” Action offers a selection of pre-defined formats from which to choose. Double-click “Set A Variable” in the Actions pane;



Highlight the Action in the Scenario Window and its associated properties can be viewed;

Variable Name	
Variable Value	
encrypted	<input type="checkbox"/>

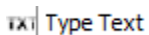
**Variable Name:** Provide a meaningful name to the variable (e.g. *loginnames* or *date*). Variable Names are case sensitive and should not include any special characters or spaces.

**Variable Value:** Select one of the pre-defined formats from the drop-down list. Options include random numbers, dates, times, and various environmental system values. For date and time pre-defined variable values, the default values are always based on current date and time. However, values can be manipulated by adding integers to, or subtracting integers from individual month, day, year, hour, minute or second parameters (e.g. *one month from today: [M+1/d/yy]*; *yesterday: [MM/dd-1/yy]*; *one hour ago: [h-1:mm:ss tt]*). Only one modification (addition or subtraction) is allowed per variable value (e.g. *[M+1/d+1/yy]* is not accepted). Do not remove brackets from the pre-defined values.

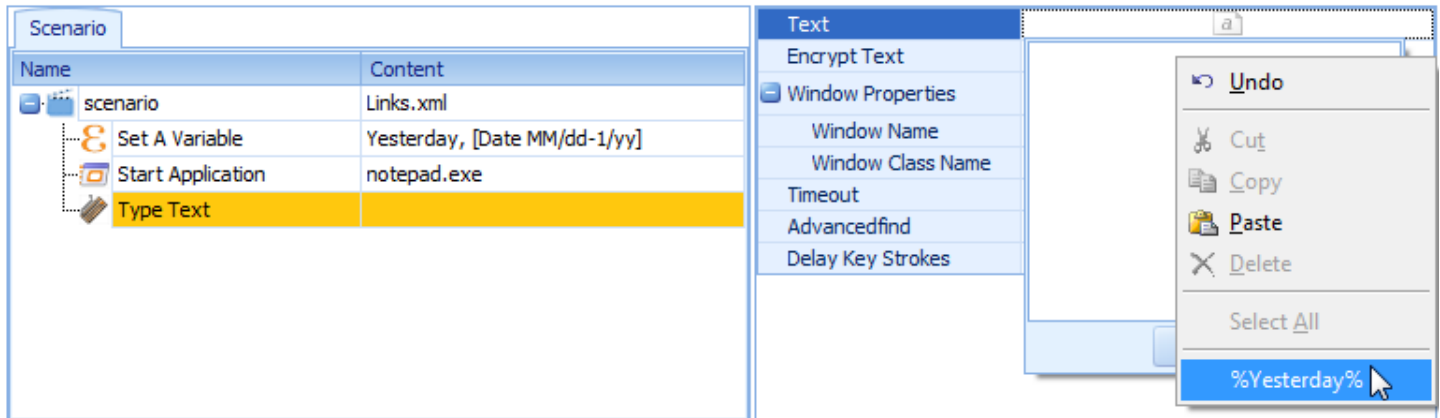
**Encrypted:** Check this box if you wish to encrypt the text for security purposes. This is useful if your variables contain passwords or other sensitive information.

Call variables into the scenario in the following manner:

From the “Keyboard” section in the Actions pane, double-click “Type Text”;

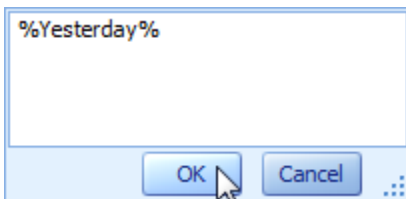


With the “Type Text” Action highlighted in Scenario Window, right click in the text field;



Select your variable from the bottom of the menu (or simply type the name of your variable, framed with percentage (%) signs, into the text field).

Click OK to complete your “Type Text” Action;



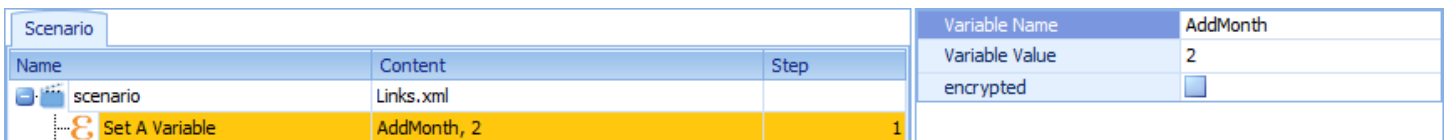
The above example opens Notepad and types yesterday's date;



## INCREMENT A VARIABLE

ScenarioBuilder can increment a variable so that each time through a loop, the scenario increases a value by a specified amount. The following example demonstrates how to increment a date by three months at a time. To add to the challenge, the first date in the sequence has to be two months from the current date.

Start by setting a variable that will provide the value by which *another* variable will be incremented. Set the “Variable Value” property to “2”;



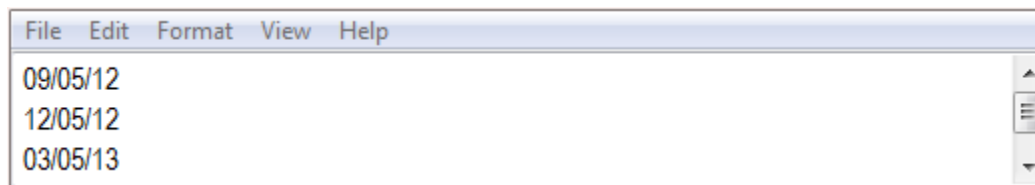
Next, the scenario opens Notepad. Then, beneath a “Loop” Action, another “Set A Variable” Action establishes a “QuarterlyMtg” date. Notice the modification which increases the “QuarterlyMtg” by the value of the “AddMonth” variable;

Scenario		
Name	Content	Step
scenario	Links.xml	
Set A Variable	AddMonth, 2	1
Start Application	notepad.exe	2
Loop	3	3
Set A Variable	QuarterlyMtg, [Date MM+%AddMonth%/dd/yy]	4

Lastly, the scenario uses “Increment A Variable” to establish the amount by which to increment the “QuarterlyMtg” date. The full scenario looks like this;

Scenario		
Name	Content	Step
scenario	Links.xml	
Set A Variable	AddMonth, 2	1
Start Application	notepad.exe	2
Loop	3	3
Set A Variable	QuarterlyMtg, [Date MM+%AddMonth%/dd/yy]	4
Type Text	%QuarterlyMtg%	5
Function Keys	Press, Enter	6
Increment A Variable	AddMonth, 3	7

The first pass through the loop will type a date two months from today into Notepad (the “AddMonth” “Variable Value” property establishes the initial value at “2”). Before jumping to the top of the loop, the “Increment A Variable” Action operates on the “AddMonth” variable by adding “3” to it (established by the “Variable Increment” property field). Now, next time through the loop, the “AddMonth” value is “5” resulting in the scenario typing a date five months from today. And finally, the last pass through the loop enters a date eight months from today into Notepad.




## EXECUTION STATUS VARIABLES

With an “Execution Status Variable” you can direct a scenario down different paths, depending on the success or failure of an Image or a Window Action. Execution Status Variables are established in the Image or Window Action properties pane and are named by default (the default value can be kept or overwritten);

Execution Status Variable	ExecMyImageStatus
---------------------------	-------------------

An Execution Status Variable's result (**success**; **failure**) can be used in conditional statements with a comparative operator (**==** (*equal*); **!=** (*not equal*)) to direct scenario behavior.

The following example uses a "Find Image" Action with a 5 second "Timeout" to look for the "Google" logo after launching [www.google.com](http://www.google.com). The "Find Image" Action has an Execution Status variable called "FoundGoogle";

Images	
Find Image #1	
Timeout	5
Continue On Failure	<input type="checkbox"/>
Tolerance	0
Black and White Contrast Ratio	0
Anchor Position X	Center
Anchor Position Y	Center
Sleep Time	1000
Image Search Area	
Wait for Image to Disappear	<input type="checkbox"/>
Execution Status Variable	FoundGoogle

The scenario uses an "If" Action that, when the site launches and displays the logo within 5 seconds, opens Notepad and types "Google site loaded within 5 seconds". Or, if the site fails to load in a timely fashion, opens Notepad and types "Fail" (Note the syntax in the example: `%FoundGoogle%==success`).


The scenario looks like this;

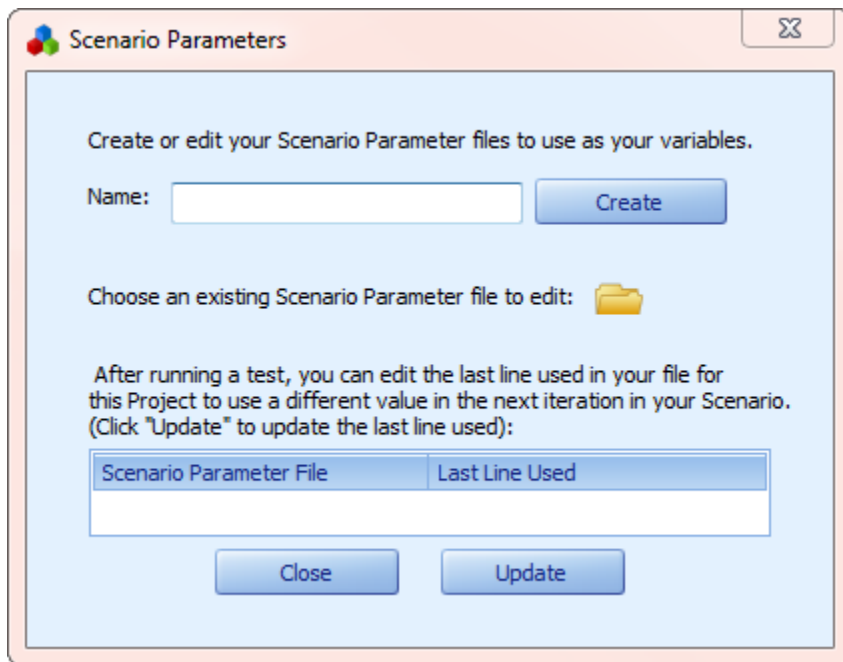
Scenario		
Name	Content	Step
scenario	LoadGoogle.xml	
Open with File Association	http://www.google.com	1
Find Image	GoogleLogo.bmp	2
If	%FoundGoogle%==success	3
then		4
Start Application	notepad.exe	5
Type Text	Google site loaded within 5 seconds	6
else		7
Start Application	notepad.exe	8
Type Text	Fail	9

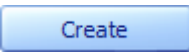
## USING A CSV FILE TO STORE VALUES FOR SCENARIO PARAMETERS

To genuinely replicate real world impact on an application, parameterizing data is critical. In ScenarioBuilder you can access unlimited rows of data values, stored in comma separated values (CSV) files, to make each playback of a scenario a unique, real user experience.

### CREATE A NEW SCENARIO PARAMETER CSV FILE

Start by creating a file that contains the values to be used for Scenario Parameters. Click the “Scenario Parameter”  icon in the ScenarioBuilder tool bar. The Scenario Parameters window opens;



Provide a name for the file and click the “Create”  button in the Scenario Parameters window;

Microsoft Excel™ or other default CSV editor will open in a new window;

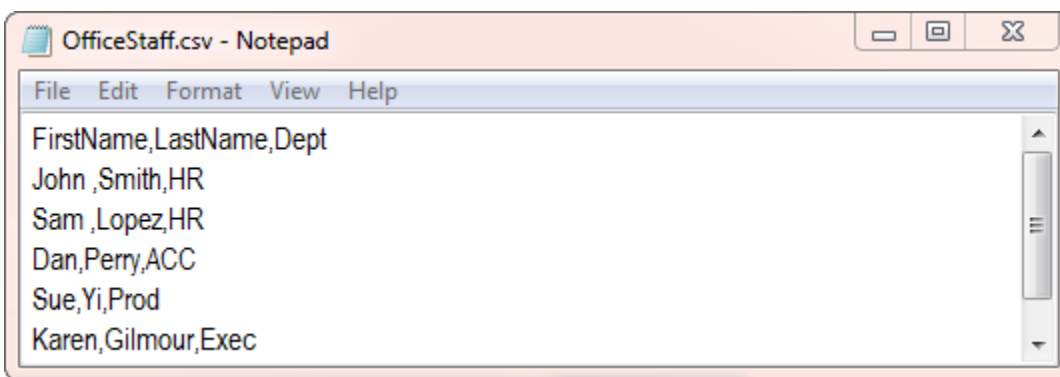
	A	B	C	D	E	F
1	ColName1	ColName2				
2						
3						
4						
5						

ScenarioBuilder gets you started by providing two column headings. Feel free to use the generic headings or overwrite them. Enter data values for the rows below the headings. Each column represents a different value to be used in a scenario. Below is a sample file containing office staff;

	A	B	C
1	FirstName	LastName	Dept
2	John	Smith	HR
3	Sam	Lopez	HR
4	Dan	Perry	ACC
5	Sue	Yi	Prod
6	Karen	Gilmour	Exec

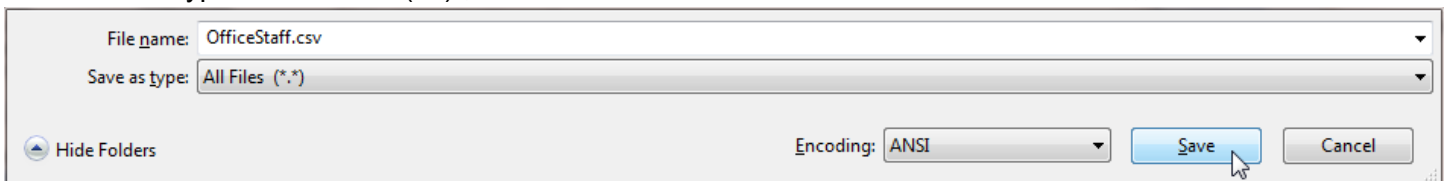
If using Excel, as in the above example, simply “Save” the file and it will be stored in CSV format in the current Project’s “Variables” folder. Exit Excel.

If using a text editor like Notepad, include commas between headers and values (no spaces);



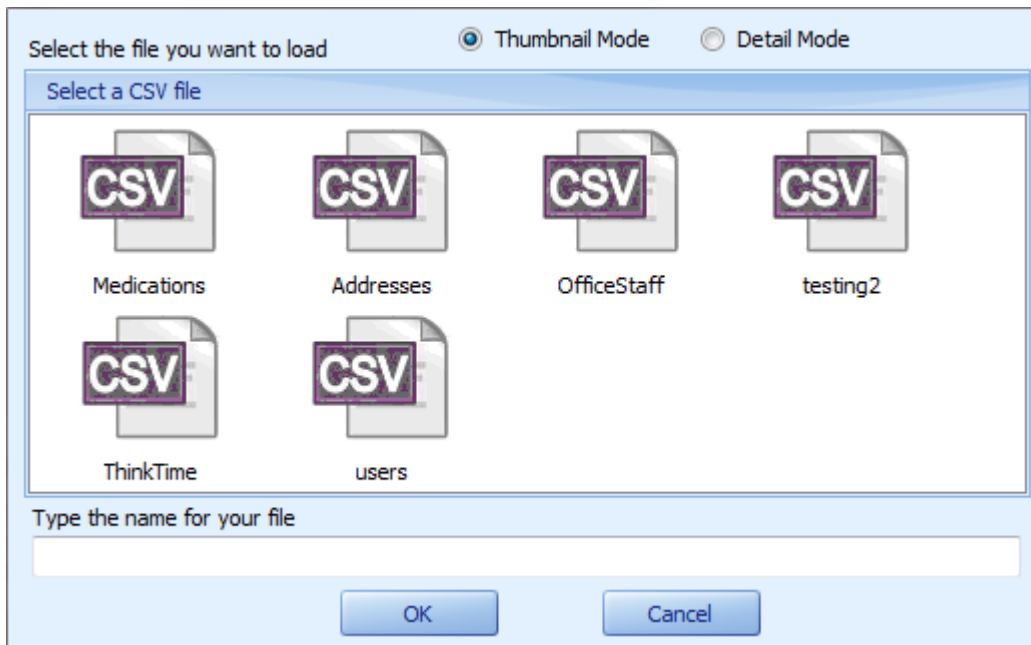
Save the file and it will be stored in CSV format in the current Project’s “Variables” folder\*. Close text editor.

\* If the editor does not automatically save the file in CSV format, use the editor’s “Save As” option and change the “Save as type” to “All Files (\*.\*)”; add the .csv extension to the file name and “Save”;



## EDIT AN EXISTING SCENARIO PARAMETER CSV FILE

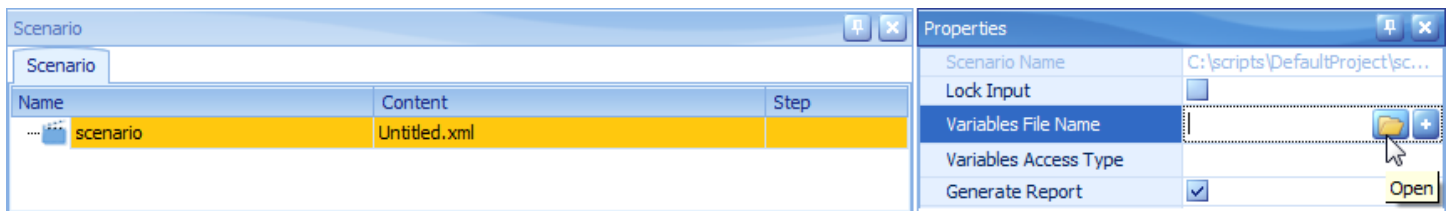
Click the folder  icon in the Scenario Parameters window to open the current Project’s “Variables” folder;



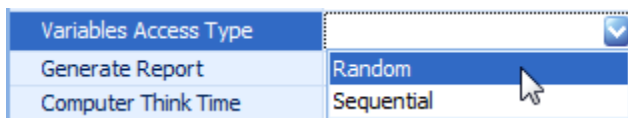
Double-click the desired file and it will open in your default CSV editor.

## ATTACH A SCENARIO PARAMETER CSV FILE


Scenario Parameter CSV files can be attached at the scenario level for global access, or at the Component level for access limited to the Component. Unless your intention is to have the scenario loop through a Component multiple times, retrieving a new row from the CSV file with each loop, attach the file at the scenario level. To do so, highlight the scenario name (top row) in the Scenario Window, and click the “Variables File Name” property. Enter the name of the desired file, or navigate to it by clicking the “Open” icon;

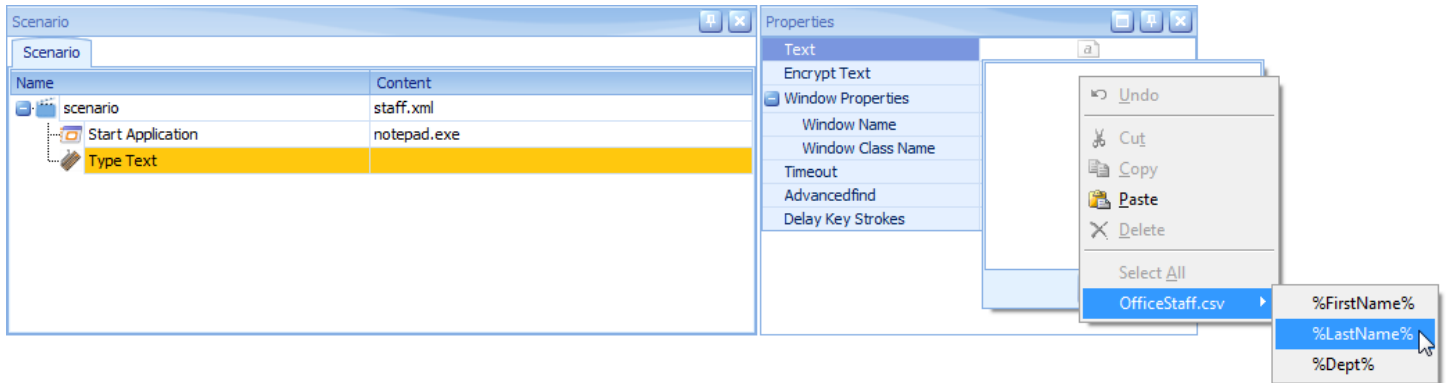


In the “Variable Access Type” field select “Random” or “Sequential” to determine how values are retrieved from the CSV file;



## ADD SCENARIO PARAMETERS

Now that you've created a CSV file and attached it to the Scenario, you're ready to add the Scenario Parameters. Add a "Type Text"  Type Text Action to the scenario. In the "Properties" pane, right click inside the "Text" field;




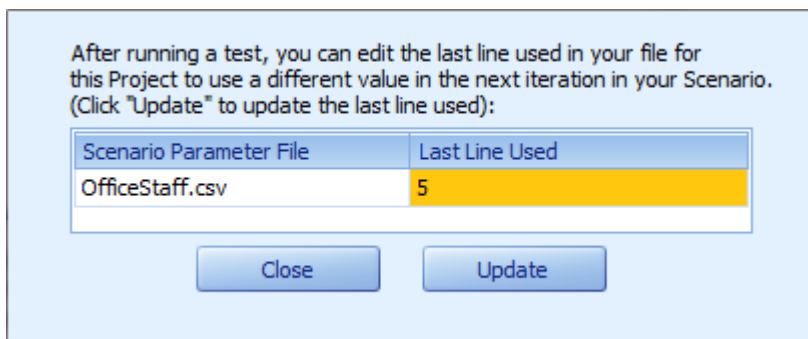
At the bottom of the pop-up menu, click the CSV file to reveal all of the fields (column headers) from the attached CSV file (file names are framed in percentage (%) signs). Select one of the fields. Typing the name of the field into the Text property field (framed with percentage signs) is also an acceptable method of adding the field.

In the preceding example, the scenario randomly retrieves a "LastName" value from the CSV file and types it into a blank Notepad document;



## LAST LINE USED IN SCENARIO PARAMETER CSV FILE

After executing a playback of the Scenario, click the "Scenario Parameters"  tool bar icon to display the "Last Line Used" from the attached CSV file;

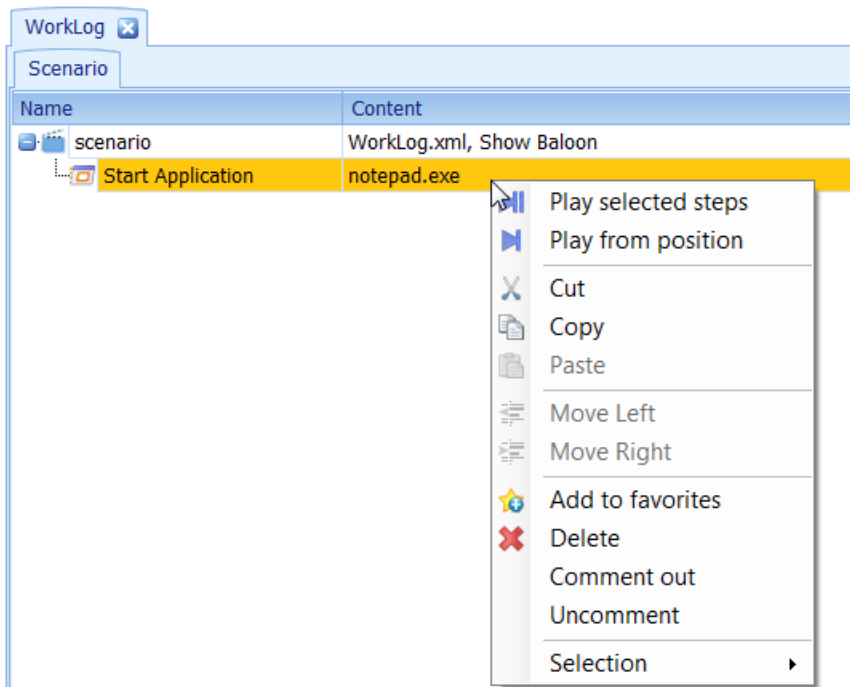


If accessing the CSV file sequentially, edit the “Last Line Used” value to stage the scenario to retrieve a desired value from the file on the next execution (if accessing randomly, adjusting the “Last Line Used” value will not impact the next playback).






Please click here for more about [Variables](#). Also, visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Use Variables Actions” video tutorial.

## ACTION CONTEXT MENU

Right click on any Action you’ve added to the Scenario Window, from either the “Tree” or “Graphical” view to see a menu of options;



<u>Icon</u>	<u>Name</u>	<u>Description</u>
	<b>Play Selected Step(s)</b>	Play the highlighted step(s) only.
	<b>Play from Position</b>	Play the scenario starting at the highlighted step.
	<b>Cut Selection</b>	Cut (copy to clipboard, and remove) the highlighted step(s).
	<b>Copy Selection to Clipboard</b>	Copy the highlighted step(s) to the clipboard.

	<b>Paste Selection from Clipboard</b>	Paste the latest “copied” item from the clipboard to the scenario.
	<b>Move Selection to the Left</b>	Move a child Action left, toward the parent level.
	<b>Move Selection to the Right</b>	Move an Action under the preceding Action, making it a child of the preceding. The child Action will only execute if the parent Action is successful.
	<b>Add to Favorites</b>	Add an Action to the Favorites folder to be called later in the current scenario or in a future scenario (from the same Project). Actions saved to Favorites include their child Actions, and they retain their Property settings.
	<b>Delete Selection</b>	Delete the highlighted step(s).
	<b>Comment out</b>	To bypass step(s) without removing from the Scenario Window, they can be “commented out”. Steps that have been “commented out” appear in italics. To hide steps that have been “commented out”, click “Show/Hide Comments” option in “View” menu.
	<b>Uncomment</b>	To re-activate “commented out” step(s), right-click on the step(s) and select “Uncomment”.
	<b>Selection</b>	Selects (highlights) all similar Actions in scenario and applies changes made to Properties across all selected Actions.

## CONDITIONAL STATEMENTS

ScenarioBuilder provides Actions that can evaluate conditions and execute user-defined reactions.

### “IF” ACTION


An “If” Action lets you set up a fork in the scenario with two branches. The scenario evaluates a result or a value, and advances down the branch determined by that result or value. The following example evaluates a user account to see if it is a “superuser”, and advances.

In this example, there is a CSV containing a list of (yes or no) values specifying whether the user account is a “superuser”;

username	superuser
jsmith	yes
gppear	no
lmoor	yes

The CSV file is attached to the Scenario;

Properties	
Scenario Name	C:\scripts\DefaultProject\scenarios\superuser.xml
Lock Input	<input type="checkbox"/>
Variables File Name	C:\scripts\DefaultProject\variables\superuser.csv
Variables Access Type	Sequential

Double-click the  If Action or drag it into the scenario pane. The scenario will use the “superuser” status as the condition that determines the path down which it advances. In the “If” Action’s Properties pane, in “Expression” field, type `%superuser%==yes`;

Properties	
Expression	<code>%superuser%==yes</code>

Then drag and drop the appropriate Actions under the “Then” and “Else” steps;

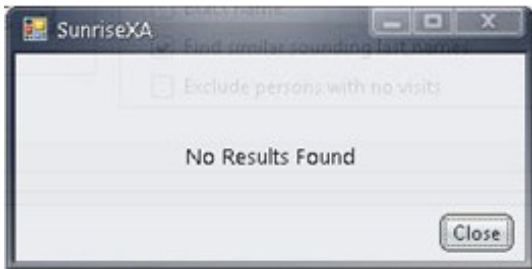
Scenario			Properties																																																				
<table border="1"> <thead> <tr> <th>Name</th> <th>Content</th> <th></th> </tr> </thead> <tbody> <tr> <td>scenario</td> <td>superuser.xml</td> <td></td> </tr> <tr> <td>  If</td> <td>%superuser%==yes</td> <td>1</td> </tr> <tr> <td>    then</td> <td></td> <td>2</td> </tr> <tr> <td>      Start A...</td> <td>notepad.exe</td> <td>3</td> </tr> <tr> <td>      Type T...</td> <td>This user is super : )</td> <td>4</td> </tr> <tr> <td>    else</td> <td></td> <td>5</td> </tr> <tr> <td>      Start A...</td> <td>notepad.exe</td> <td>6</td> </tr> <tr> <td>      Type T...</td> <td>This user is just average : (</td> <td>7</td> </tr> </tbody> </table>			Name	Content		scenario	superuser.xml		If	%superuser%==yes	1	then		2	Start A...	notepad.exe	3	Type T...	This user is super : )	4	else		5	Start A...	notepad.exe	6	Type T...	This user is just average : (	7	<table border="1"> <tbody> <tr> <td>Scenario Name</td> <td>C:\scripts\DefaultProject\...</td> </tr> <tr> <td>Lock Input</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Variables File Name</td> <td>C:\scripts\DefaultProject\...</td> </tr> <tr> <td>Variables Access Type</td> <td>Sequential</td> </tr> <tr> <td>Generate Report</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>Computer Think Time</td> <td>1000</td> </tr> <tr> <td>Exclude Think Time from report</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Maximum Execution Time</td> <td></td> </tr> <tr> <td>Reference Time</td> <td>2.014</td> </tr> <tr> <td colspan="2">Description</td> </tr> <tr> <td colspan="2">Scenario Name</td> </tr> <tr> <td colspan="2">Scenario's file name including its path.</td> </tr> </tbody> </table>		Scenario Name	C:\scripts\DefaultProject\...	Lock Input	<input type="checkbox"/>	Variables File Name	C:\scripts\DefaultProject\...	Variables Access Type	Sequential	Generate Report	<input checked="" type="checkbox"/>	Computer Think Time	1000	Exclude Think Time from report	<input type="checkbox"/>	Maximum Execution Time		Reference Time	2.014	Description		Scenario Name		Scenario's file name including its path.	
Name	Content																																																						
scenario	superuser.xml																																																						
If	%superuser%==yes	1																																																					
then		2																																																					
Start A...	notepad.exe	3																																																					
Type T...	This user is super : )	4																																																					
else		5																																																					
Start A...	notepad.exe	6																																																					
Type T...	This user is just average : (	7																																																					
Scenario Name	C:\scripts\DefaultProject\...																																																						
Lock Input	<input type="checkbox"/>																																																						
Variables File Name	C:\scripts\DefaultProject\...																																																						
Variables Access Type	Sequential																																																						
Generate Report	<input checked="" type="checkbox"/>																																																						
Computer Think Time	1000																																																						
Exclude Think Time from report	<input type="checkbox"/>																																																						
Maximum Execution Time																																																							
Reference Time	2.014																																																						
Description																																																							
Scenario Name																																																							
Scenario's file name including its path.																																																							

The first playback of the scenario retrieves the first row from the CSV file (“Variable Access Type” property was set to “Sequential”). User jsmith has a “superuser” value of “yes”, therefore;

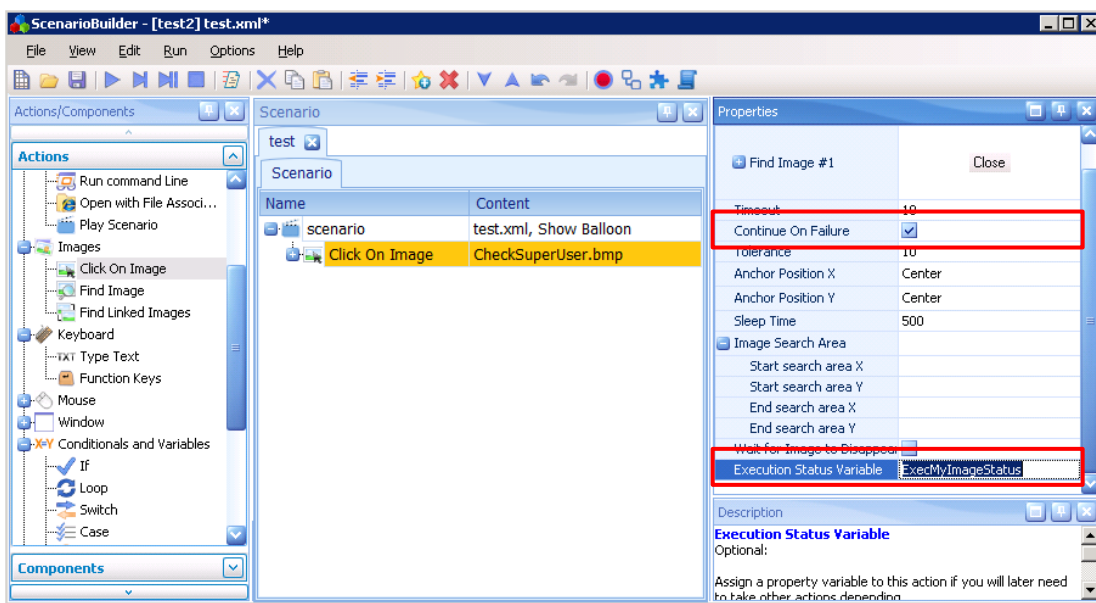
Untitled - Notepad	
File Edit Format View Help	
This user is super : )	


## “IF” ACTION WITH EXECUTION STATUS VARIABLE

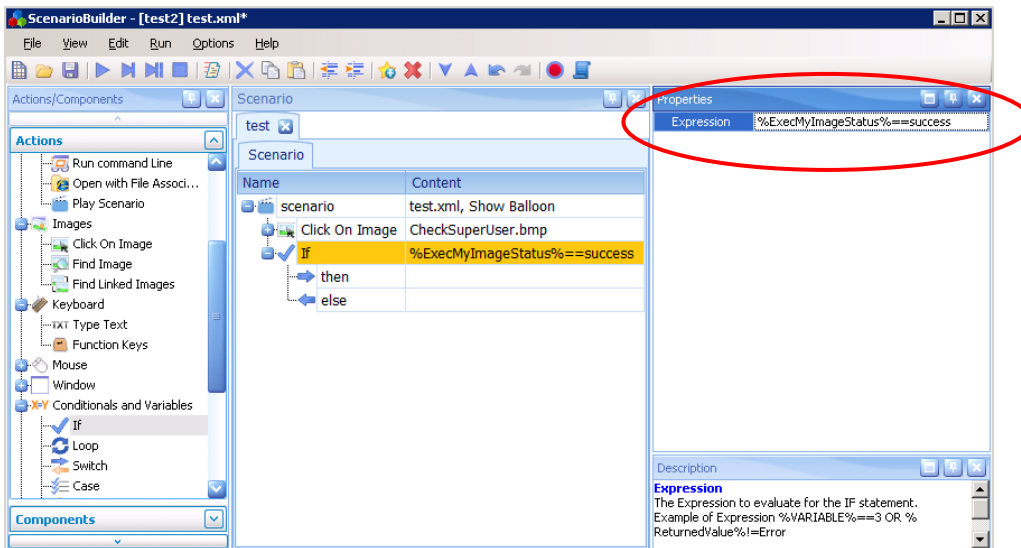
You may want to validate the success or failure of an Action and proceed according to the result. For example, you need to check for a result after executing the search function on a page, the search function might return a result or might return a popup window stating “no results found”. To test the result, take an image of the “no results found” pop-up window;



Enable the “Continue on Failure” property and note the name of the “Execution Status Variable” property (feel free to overwrite this value);



Double click on the  If Action or drag it to the scenario pane. In the “If” Action’s “Expression” property, enter a status expression for the “Execution Status Variable”, for example `%ExecMyImageStatus%==success`.



Now, drag and drop the appropriate Actions under the “If” Action’s “Success” and “Failure” branches.

A Component can be validated for success or failure in that same way as an Image Action. Enable the Component’s “Continue on Failure” property and note the name of its “Execution Status Variable” property. Then create an “If” statement based on the success and failure of the Component.

Visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Conditional Statements” video tutorial.

## CASE/SWITCH

The Case and Switch Actions allow you to build complex scenarios that will respond in a variety of ways, depending on the result of a value. Where the “If” Action can evaluate a condition as being either true or false, and can provide a branch for either possibility, the Case/Switch Actions allow for virtually unlimited result possibilities, and can provide appropriate courses of action for all possibilities.

For example, a web form in which students rate their teacher has 4 possible responses. In order to test the form as real users would, you need a scenario that randomly selects one of the 4 responses;


**2. Please rate your overall satisfaction with the Professor/Course Instructor:**

☐ Very Dissatisfied

☐ Dissatisfied

☐ Satisfied

☐ Very Satisfied

Start by creating a Scenario Parameters file in ScenarioBuilder. Click the “Scenario Parameters”  icon in the toolbar to open the “Scenario Parameters” window. Provide a name for the file and click “Create”;

Create or edit your Scenario Parameter files to use as your variables.

Name:

Create a column named “rank” and add values 1-4;

File	Edit	Format	View	Help
rank				
1				
2				
3				
4				


Save the file and attach it to the scenario. Set the “Variable Access Type” property to “Random”;

Properties	
Scenario Name	CourseRanking.xml
Lock Input	<input type="checkbox"/>
Variables File Name	C:\scripts\DefaultProject\variables\rank.csv
Variables Access Type	Random

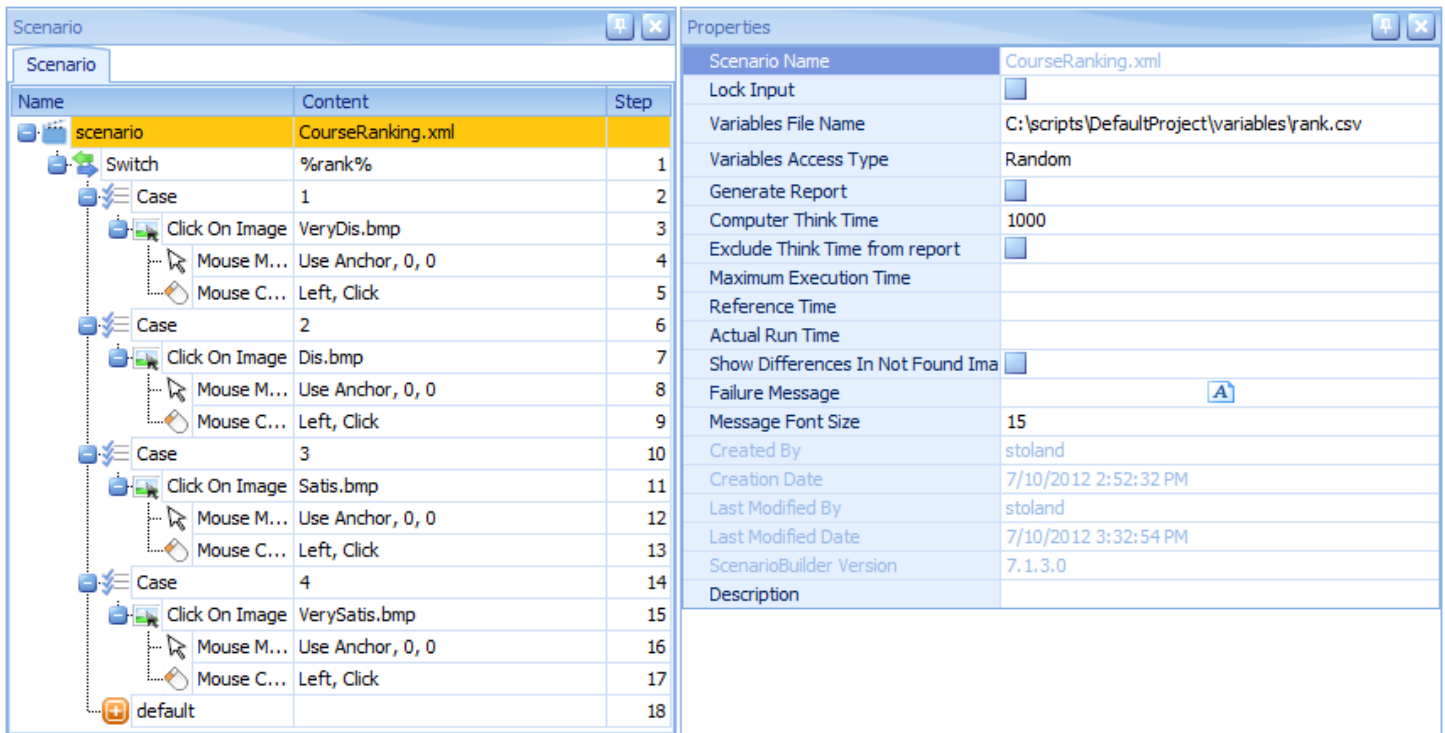
Add a “Switch” Action to the scenario and set the “Variable Name” property to “%rank%”. Be sure to include the percentage signs framing the variable name. This is the value that will determine the Cases and the subsequent Actions associated with each Case. Notice a “Case” child Action came into the Scenario Window with the “Switch” Action. Click on the “Case” and set the “Variable Value” property to 1. Drag a “Click on Image” Action under the “Case” row and capture an image of the “Very Dissatisfied” choice from the web form. Here’s what you have so far;

Name	Content
scenario	CourseRanking.xml
Switch	%rank%
Case	1
Click On Image	VeryDis.bmp
Mouse Move	Use Anchor, 0, 0
Mouse Click	Left, Click
default	

Properties	
Images	
Find Image #1	 Very D
Timeout	10
Continue On Failure	<input type="checkbox"/>
Tolerance	0
Black and White Contrast Ratio	0
Anchor Position X	left
Anchor Position Y	Center
Sleep Time	1000

Add 3 more “Case” Actions below the Switch, using values 2-4, and associate each with one of the choices on the form. Ignore the “default” row, which can be used as a “catch all” for values that don’t correspond to a Case in the scenario. Here’s what the final scenario looks like;



The screenshot shows the ScenarioBuilder interface with two panes: Scenario and Properties.

**Scenario Pane:**

Name	Content	Step
scenario	CourseRanking.xml	
Switch	%rank%	1
Case	1	2
Click On Image	VeryDis.bmp	3
Mouse M...	Use Anchor, 0, 0	4
Mouse C...	Left, Click	5
Case	2	6
Click On Image	Dis.bmp	7
Mouse M...	Use Anchor, 0, 0	8
Mouse C...	Left, Click	9
Case	3	10
Click On Image	Satis.bmp	11
Mouse M...	Use Anchor, 0, 0	12
Mouse C...	Left, Click	13
Case	4	14
Click On Image	VerySatis.bmp	15
Mouse M...	Use Anchor, 0, 0	16
Mouse C...	Left, Click	17
default		18

**Properties Pane:**

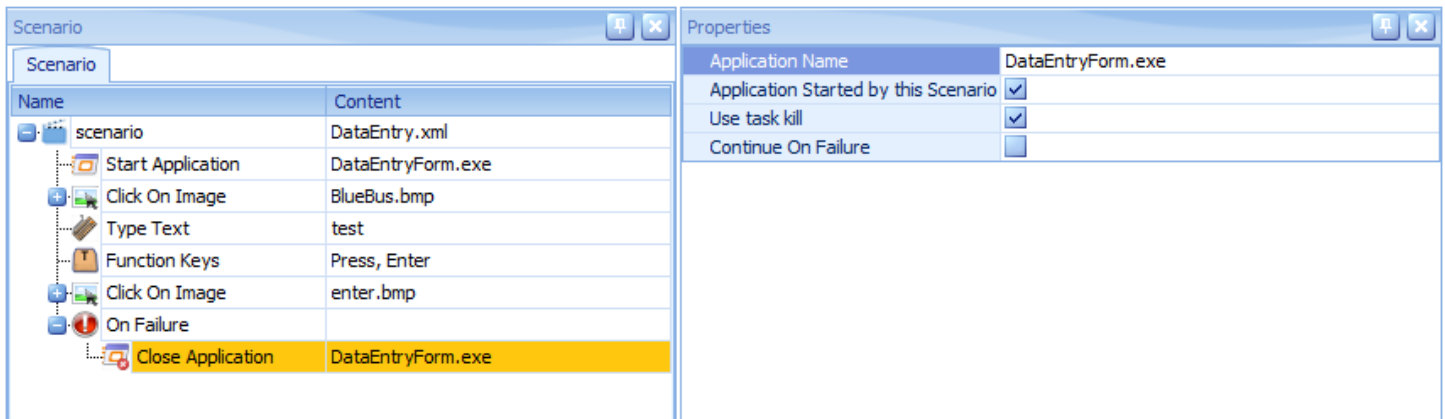
Property	Value
Scenario Name	CourseRanking.xml
Lock Input	<input type="checkbox"/>
Variables File Name	C:\scripts\DefaultProject\variables\rank.csv
Variables Access Type	Random
Generate Report	<input type="checkbox"/>
Computer Think Time	1000
Exclude Think Time from report	<input type="checkbox"/>
Maximum Execution Time	
Reference Time	
Actual Run Time	
Show Differences In Not Found Images	<input type="checkbox"/>
Failure Message	<input type="text" value="A"/>
Message Font Size	15
Created By	stoland
Creation Date	7/10/2012 2:52:32 PM
Last Modified By	stoland
Last Modified Date	7/10/2012 3:32:54 PM
ScenarioBuilder Version	7.1.3.0
Description	

The resulting scenario makes a random choice, clicking on one of the four evaluation rankings, depending on the value retrieved from the Scenario Parameters file.

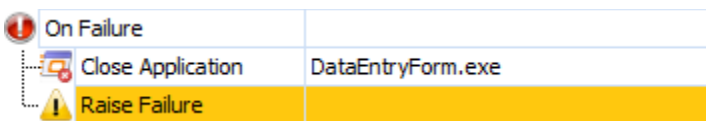
For more about Case/Switch, visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Conditional Statements” video tutorial.

## ON FAILURE

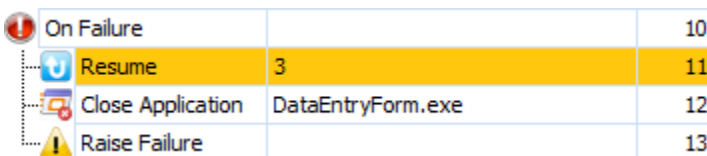
The “On Failure” section of a scenario determines what Action(s) to take when a scenario or Component fails. Add the “On Failure” Action to the end of the scenario and drag the appropriate Actions underneath it. In the event of a failure, the scenario will skip to the “On Failure” section and execute the Actions. “On Failure” can be as simple as closing the application, thereby resetting the desktop for another iteration of the scenario (see below), or as complicated as undoing multiple changes made by the scenario up to the failed step.



When a scenario resorts to the “On Failure” steps, it does not report the execution as a failure, *unless* a “Raise Failure” Action is included in the “On Failure” section;




Add a “Resume” Action to the “On Failure” section to retry the scenario from the point of the failure;

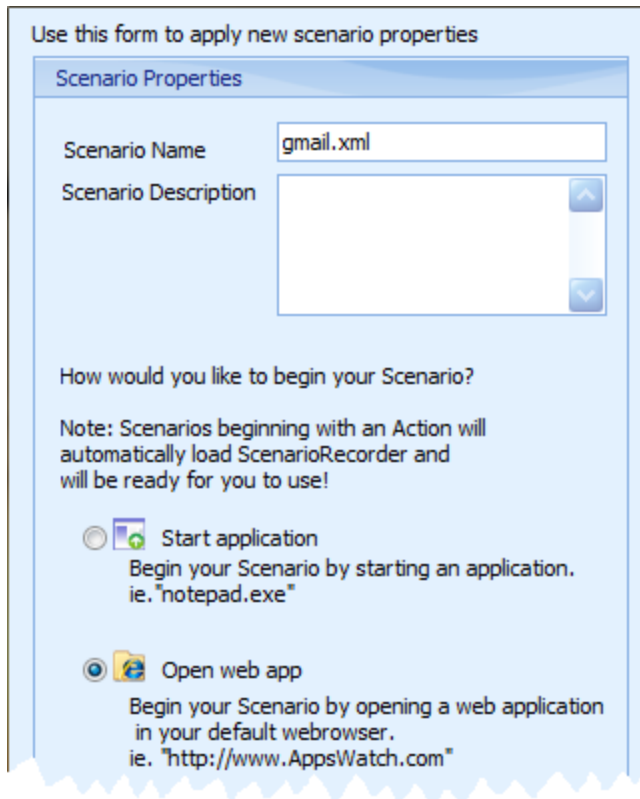


Set the “Retries” value to establish how many times to attempt the “Resume”. After the retries are exhausted, the scenario will advance through the subsequent steps of the “On Failure” section.

For more about “On Failure”, visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> to view our “Using On Failure Action” video tutorial.

## SAMPLE ADVANCED SCENARIO

The following example demonstrates a scenario that creates a Gmail account, with an emphasis on Scenario Recorder. Start by opening ScenarioBuilder and clicking the “New Scenario”  [New Scenario](#) link on the splash screen. The Scenario “Properties” form opens;



Use this form to apply new scenario properties


**Scenario Properties**


Scenario Name:

Scenario Description:

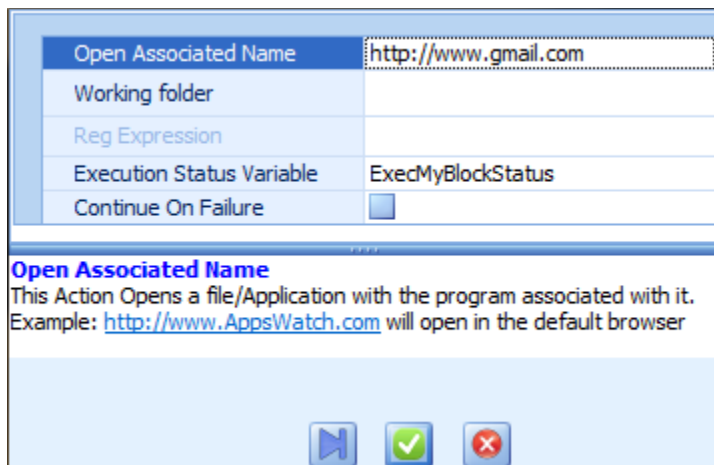
How would you like to begin your Scenario?

Note: Scenarios beginning with an Action will automatically load ScenarioRecorder and will be ready for you to use!

☐  Start application  
Begin your Scenario by starting an application.  
ie. "notepad.exe"




☐  Open web app  
Begin your Scenario by opening a web application  
in your default webbrowser.  
ie. "http://www.AppsWatch.com"

Choose “New Scenario” and name the new scenario “Gmail.” Select “Open Web App” and click “OK” to launch the “Quick Add” menu;



Open Associated Name	<input type="text" value="http://www.gmail.com"/>
Working folder	
Reg Expression	
Execution Status Variable	ExedMyBlockStatus
Continue On Failure	<input type="checkbox"/>

**Open Associated Name**  
This Action Opens a file/Application with the program associated with it.  
Example: <http://www.AppsWatch.com> will open in the default browser

On the “Quick Add” menu, in the ‘Open Associated Name’ field, enter the Gmail URL: <http://www.gmail.com>.

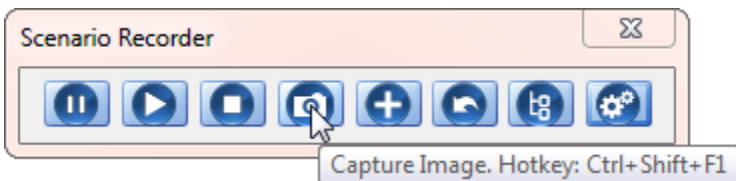
Press <Enter> to playback and verify this Action. (Note: Actions can be invoked by user-defined Hot-Keys that can be set in Scenario Recorder *Preferences*). Gmail launches in the default browser and the Scenario Recorder panel docks in the lower right hand corner of the screen;



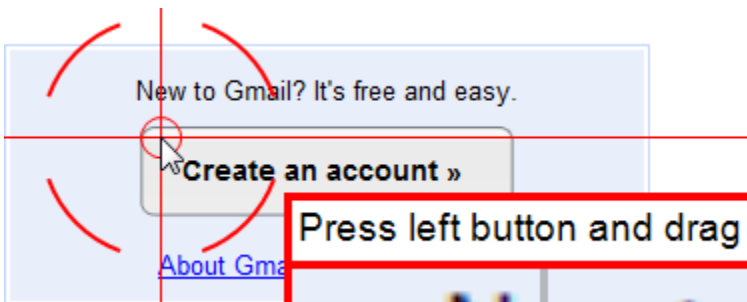
Scenario Recorder allows you to create scenarios on the fly, recording your key strokes and simplifying common Actions like *Image Capture* and *Begin* and *End Transaction*. The Scenario Recorder generates the same results as ScenarioBuilder, logging a step-by-step list of Actions which can be edited from the graphical or tree views in ScenarioBuilder. Complex Actions like conditional statements, switches or “On Failure” commands can be added after the initial recording is completed.

Scenario Recorder DOES NOT record mouse movements or think time.

Next, we’ll need to click the “Create an Account” button. Click the ‘Capture Image’ button on the Scenario Recorder;



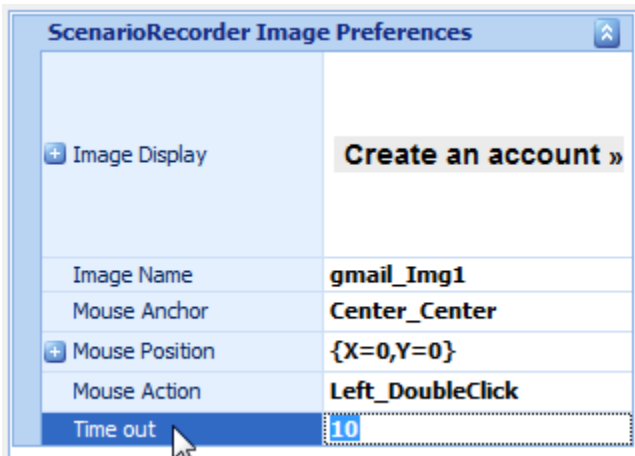
Capture a select area of the button;



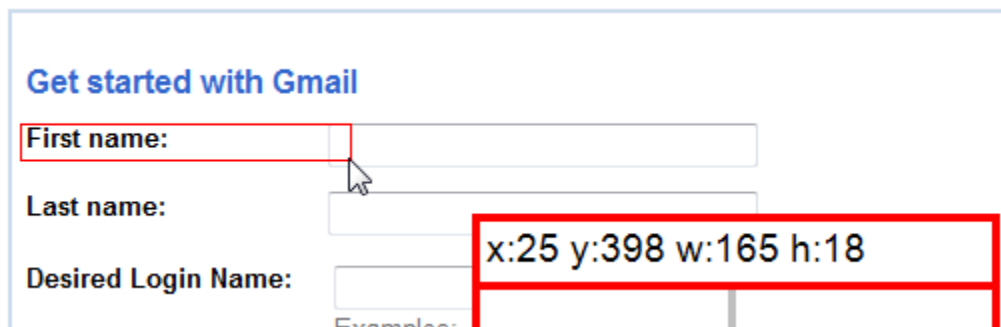
As a rule, it is not necessary to capture an entire image. Simply capture a unique portion of an image and ScenarioBuilder will recognize it. During playback, your scenario will exhaust the allotted “Timeout” period before failing to find an image. Be sure to allow an adequate timeout for the image to load and render on the page. ScenarioBuilder’s advanced search technology finds images wherever they appear on the desktop.

Every time you capture an image while in Record mode, an image preferences menu opens; displaying the properties of the Image Capture Action you’ve just recorded (Image Name, Mouse Anchor, Mouse Position,

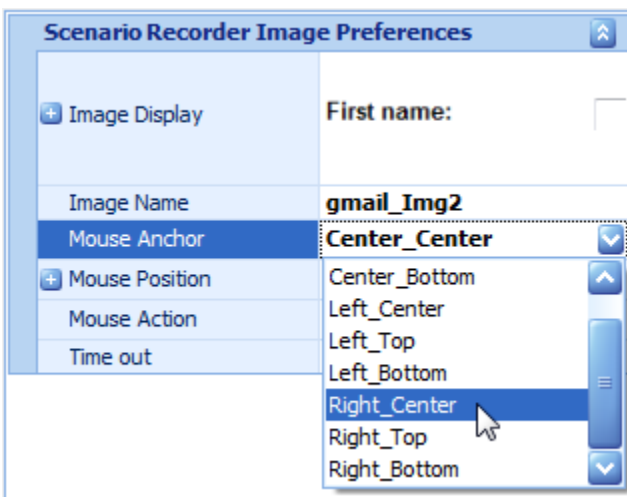
Mouse Action and Time out). Adjust these settings or accept the defaults (Note: set your own default preferences by clicking on Scenario Recorder *Preferences* button on the Scenario Recorder menu);



Press <Enter> after setting the preferences for the captured image. The Gmail website advances to the page that will let you fill out a form to create an account. Locate the cursor in the “First Name” field with the Capture Image action. Include the field name as well as a little section of the text field in the image;



Set the “Mouse Anchor” to Right\_Center so the cursor will land in the text field, poised for the next step;



This next series of steps shows where the Scenario Recorder is a real time saver. Simply type text and tab through the form. Scenario Recorder will record each keystroke in real time.

### Get started with Gmail

**First name:**

**Last name:**

**Desired Login Name:**  @gmail.com  
Examples: JSmith, John.Smith

**Choose a password:**  [Password strength:](#)

Good

  
Minimum of 8 characters in length.

**Re-enter password:**

☒ Stay signed in

☒ Enable Web History [Learn More](#)

**Default Homepage** ☒ Set Google as my default homepage.  
Your default homepage in your browser is the first page that appears when you open your browser.

**Security question:**  ▼  
If you forget your password we will ask for the answer to your security question. [Learn More](#)

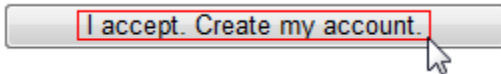
**Answer:**


**Recovery email:**   
This address is used to authenticate your account should you ever encounter problems or forget your password. If you do not have another email address, you may leave this field blank. [Learn More](#)

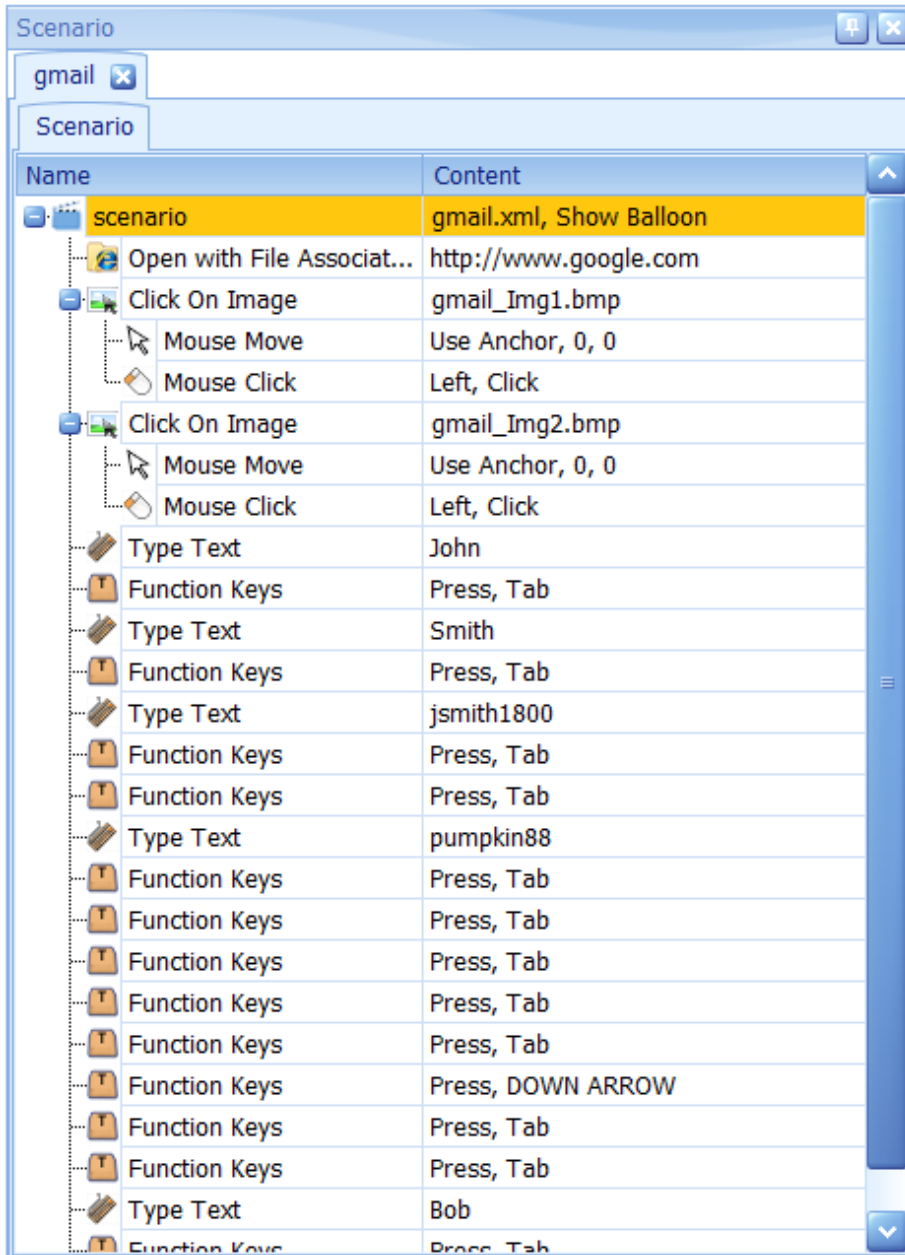
**Location:**  ▼

**Birthday:**   
MM/DD/YYYY (e.g. "6/24/2011")

Finish the scenario by capturing the 'I Accept. Create my account.' button at the bottom of the form;





Click the “Stop”  button to end recording return to the ScenarioBuilder standard view where all of the recorded steps have been logged;



To view a video demonstrating the making of the above referenced Scenario, please visit <http://www.nrgglobal.com/Support-Overview/scenariobuilder-training-videos> and click on the “Scenario Recorder Demo” link.

## SCENARIO PROPERTIES

Scenario “Properties” establish the playback settings for each scenario. To edit the scenario properties, click the top row in the Scenario Window. The scenario properties display in the “Properties” pane;

Properties	
Scenario Name	gmail2.xml
Lock Input	<input type="checkbox"/>
Variables File Name	
Variables Access Type	
Generate Report	<input checked="" type="checkbox"/>
Computer Think Time	1000
Exclude Think Time from report	<input type="checkbox"/>
Maximum Execution Time	
Reference Time	
Actual Run Time	
Show Differences In Not Found Images	<input type="checkbox"/>
Show Balloon	<input checked="" type="checkbox"/>
Failure Message	
Message Font Size	15
Created By	stoland
Creation Date	2/4/2013 4:05:53 PM
Last Modified By	stoland
Last Modified Date	2/4/2013 4:05:53 PM
ScenarioBuilder Version	7.5.0.0
Description	

**Scenario Name:** The name of the scenario.

**Lock Input:** If this field is checked, the keyboard and mouse will be locked during the execution of the scenario.

**Variables File Name (optional):** If using Scenario Parameters from a CSV file, attach the CSV file here. Click the folder icon to browse for the desired CSV file. Refer to the preceding section, [Scenario Parameters](#), for more information about working with variables.

**Variables Access Type (optional):** Determines whether the variable fields in the CSV file will be played back sequentially or randomly.

**Generate Report:** Check this field if you would like a report generated for the playback.

**Computer Think Time:** Specify the time interval between each step of execution. Values are in milliseconds; the default is 1000 milliseconds.

**Exclude Think Time from report:** If Checked, the 'Computer Think Time' set above will be excluded from the reported execution time.

**Maximum Execution Time:** Set the time allowed for the scenario to complete its playback. This parameter is useful to determine if applications are responding in a timely manner. If the playback does not complete within the time specified, the scenario will fail and a screenshot will be taken of the failed scenario step.

**Reference Time:** This field is used as the standard to which subsequent iterations of the scenario are compared.

**Actual Run Time:** The complete duration of the scenario from start to end. “Actual Run Time” does not factor out “Think Time” or response time of steps suspended by “Timer” Actions (as compared to “Reference Time” which does exclude these values).

**Show Differences in Not Found Image:** Check this field if you would like to see the differences between the not found images.

**Show Balloon:** If checked, the step-by-step scenario status balloon will display for each rUser session in AppLoader, and for each ScenarioStation in AppsWatch.

**Failure Message:** This is the message generated when the scenario fails. You can customize the message by using the following variables:

%\_error%: default message

%\_Scenario%: Failed scenario Name

%\_Component%: Failed Component Name

%\_step%: Failed Step Number

%\_Action%: Failed Action

%\_type%: Failure Type (test or system)

**Message font size:** Set the failure message font size (Use a smaller font size for long messages).

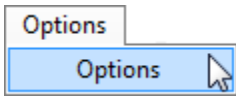
The remaining fields (Created By, Creation Date...) are read-only and show the general properties of your scenario file. This information is automatically populated by the system.

## PROJECTS

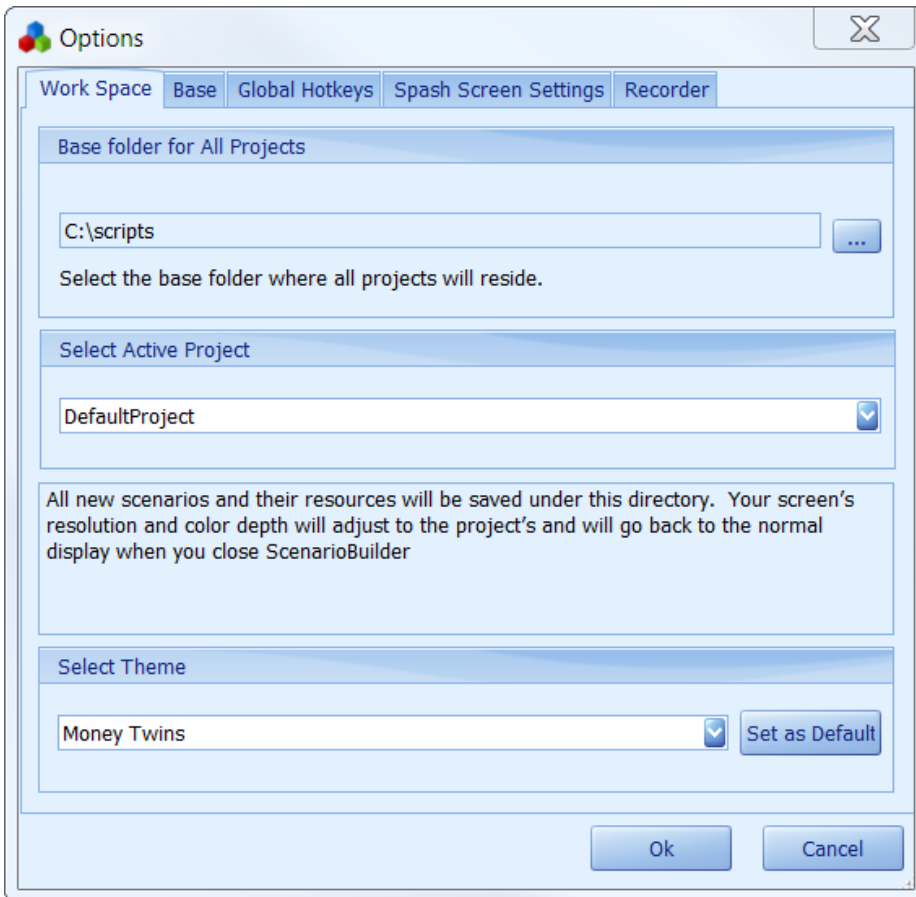
Upon installation, ScenarioBuilder creates the “Default Project” folder, where scenarios will be stored *unless* a new Project is created and “Switched To”. Creating a Project for each business application being tested not only keeps test cases organized, but allows you to share resources among related scenarios.

## OPTIONS


To make changes to ScenarioBuilder’s Work Space, click “Options”;

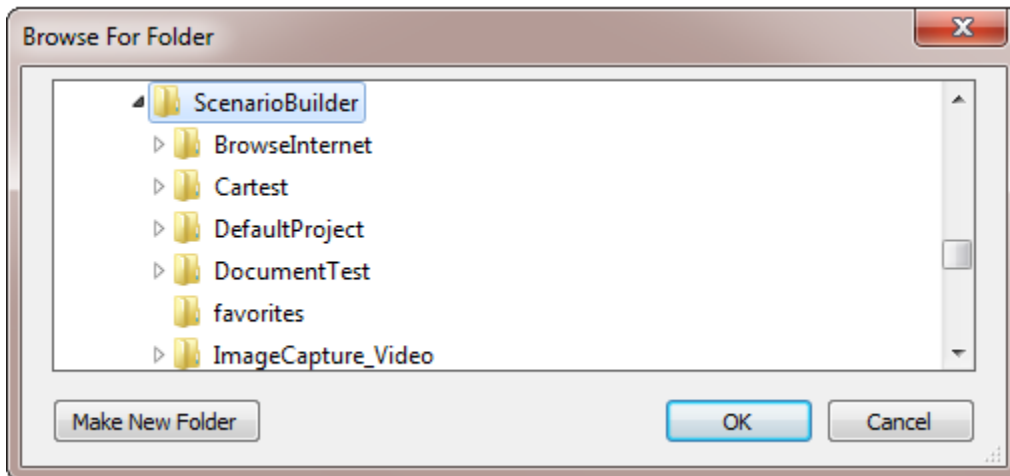


The “Options” menu opens;



## WORK SPACE

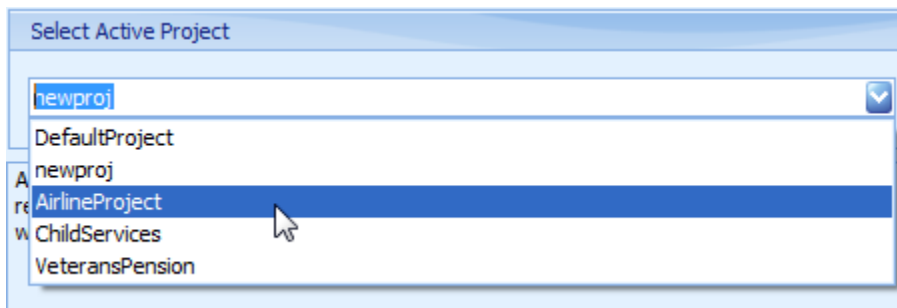
In the “Options” window, with the “Work Space” tab enabled, click the ellipses  button in the “Base folder for All Projects” pane to navigate to the desired Project base folder;



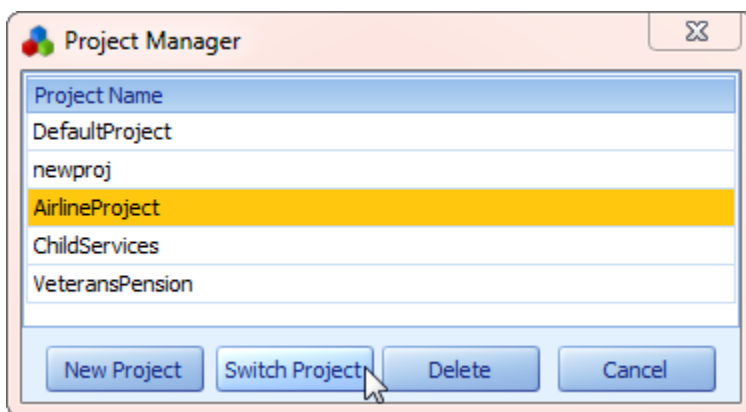
Highlight the desired folder and click “OK” to save changes. Projects created from this point forward will be stored in the selected location (Note: this change does not relocate previously created Projects).

## CHANGE ACTIVE PROJECT

ScenarioBuilder lets you change the Active Project, either by selecting a Project from the drop-down field in the “Select Active Project” pane of the “Options” window;

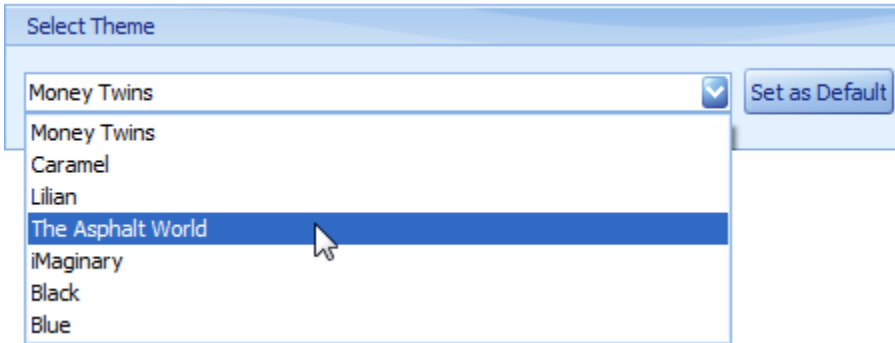


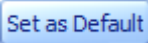
Or, from the Project Manager, by highlighting a Project and clicking the “Switch Project” button;

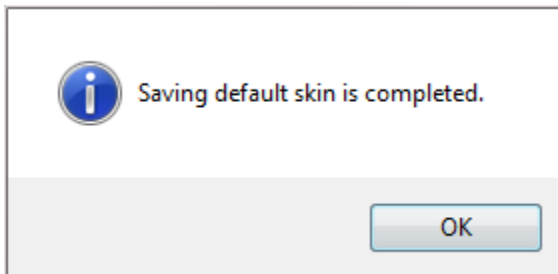


## CHANGE THEME

Themes establish the “skin” of the ScenarioBuilder session. To change the Theme, in the “Select Themes” pane of the “Options” window, use the drop down field to select a “Theme”;

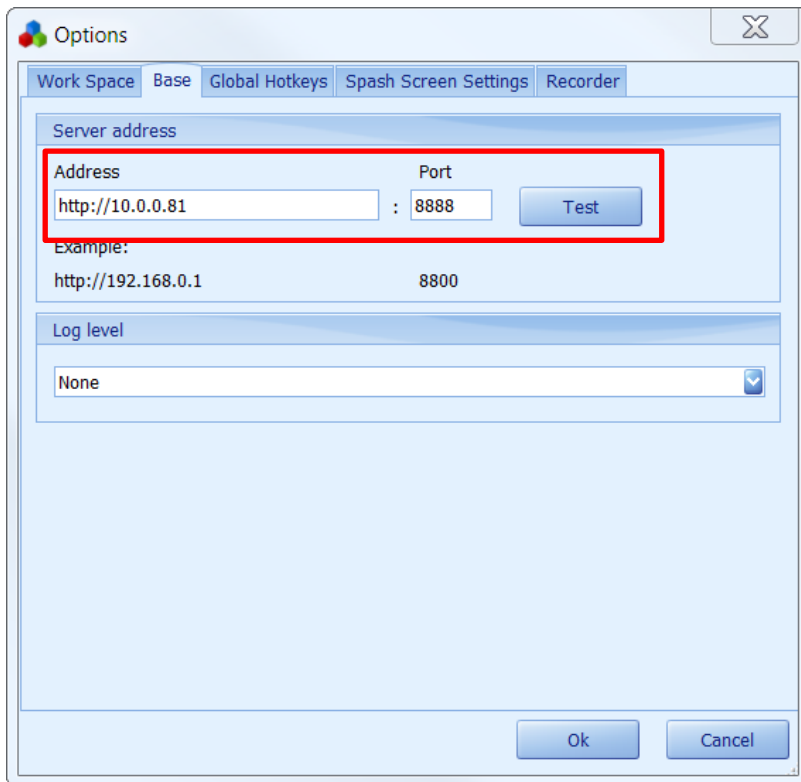


Click “Set as Default”  to have ScenarioBuilder use the Theme every time it opens;

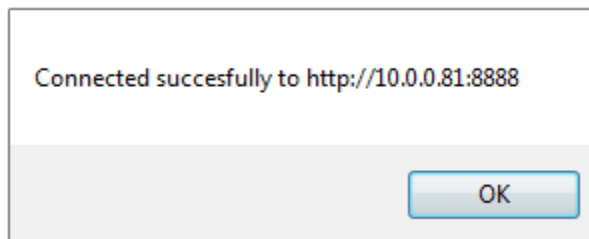


## BASE

Change the Base (AppsWatch) or Controller (AppLoader) to which ScenarioBuilder “sends” scenarios. In the “Options” menu, with the “Base” tab enabled, enter the IP address and Port of the desired Base or Controller;



Click the “Test”  button to ensure proper configuration;



This setting establishes the destination for scenarios when the “Send Scenario to Controller”

 menu option or  tool bar option is used.

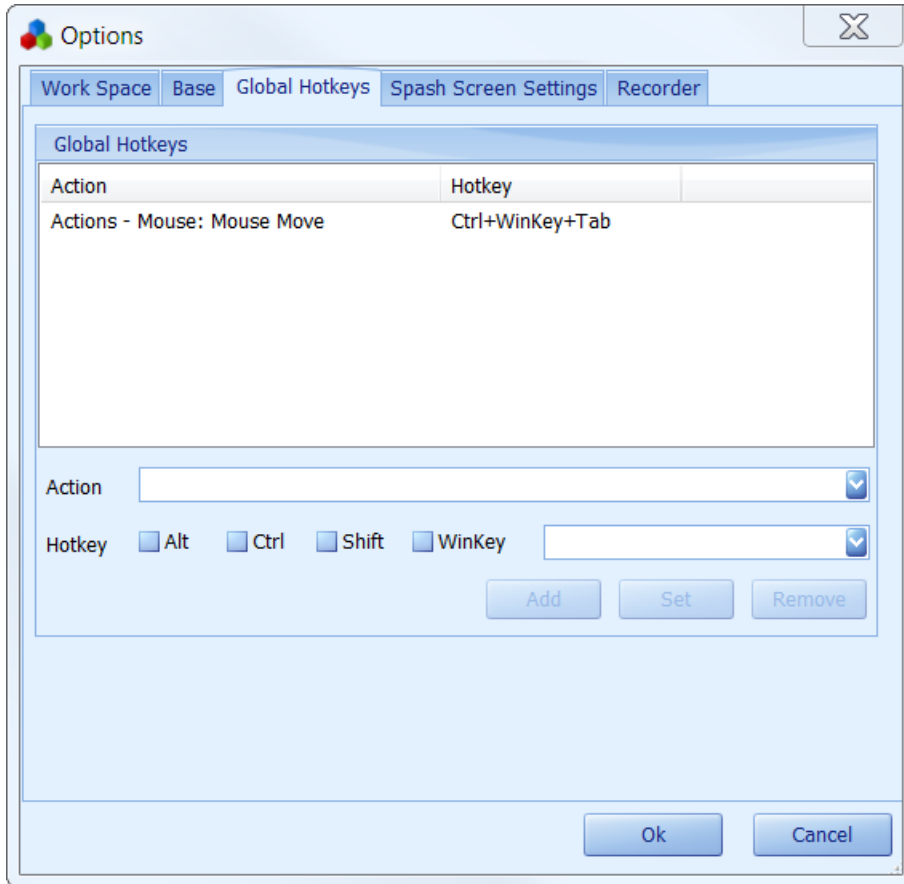
## CHANGE LOG LEVEL

In the “Options” menu, “Base” tab, use the “Log Level” drop-down field to change the log level;



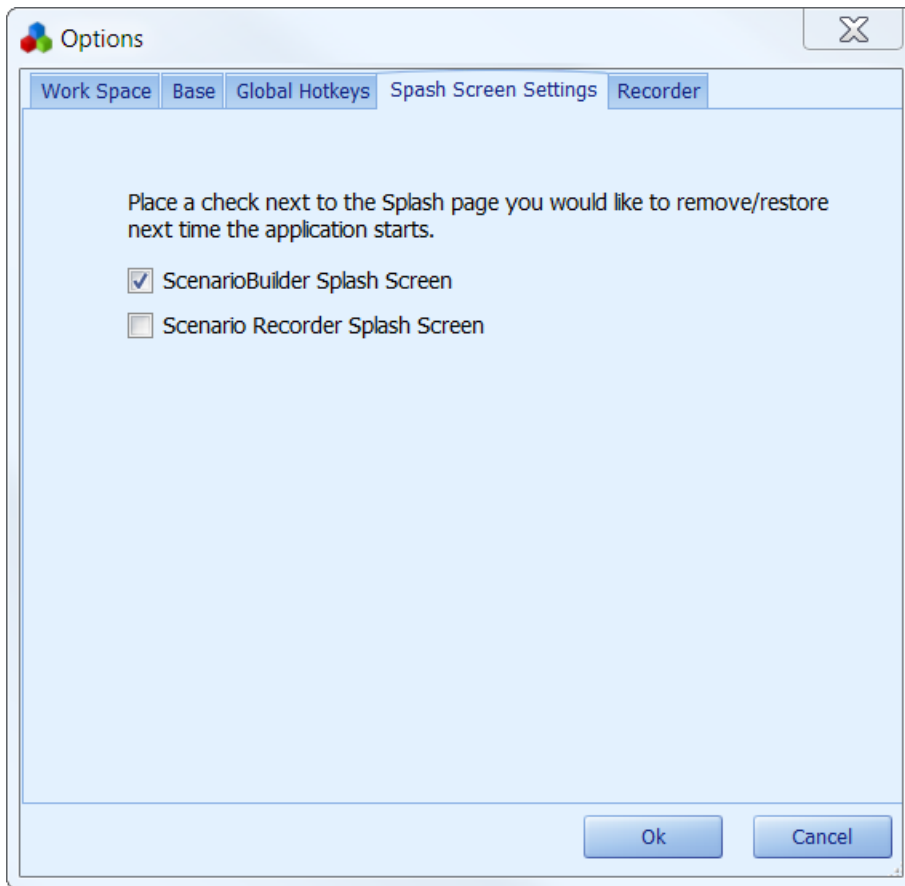
## GLOBAL HOTKEYS

Manage hotkeys that trigger Actions or menu options in the Scenario Recorder and ScenarioBuilder standard view. A predefined set of Scenario Recorder hotkeys have been configured. These can be edited....



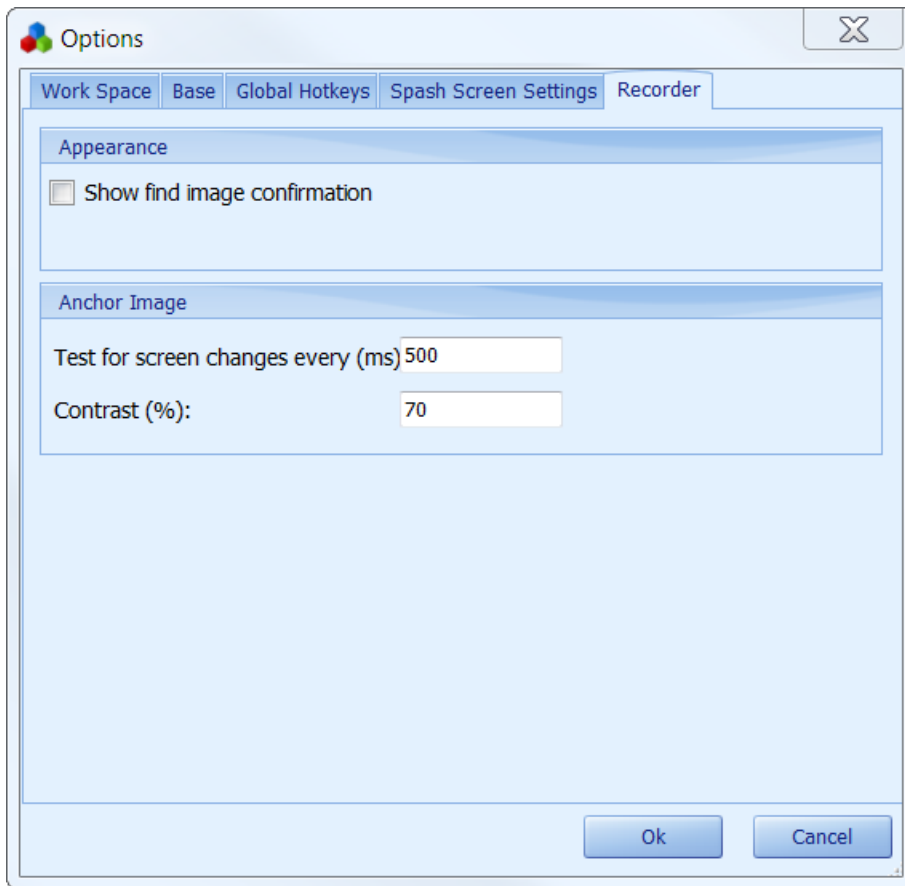
## SPLASH SCREEN SETTINGS

Select your Splash Screen preferences for ScenarioBuilder and Scenario Recorder. Enable or disable the Splash Screen by ticking the appropriate checkbox.



## RECORDER

Set the Scenario Recorder preferences from the “Recorder” tab of the Options menu.



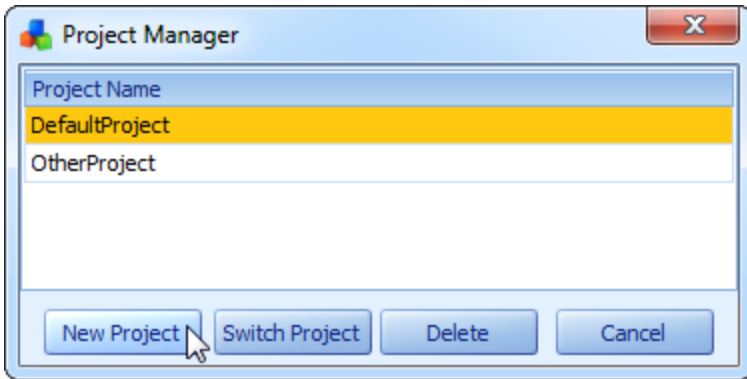
**Show find image confirmation:** If enabled, a pop-up confirmation window occurs after each image capture while in Scenario Recorder.

**Test for screen changes every (ms):** During the recording process, Scenario Recorder checks the screen for the current anchor image, alerting the user when the anchor image has changed or is no longer present. Use this setting to establish how often Scenario Recorder checks for changes.

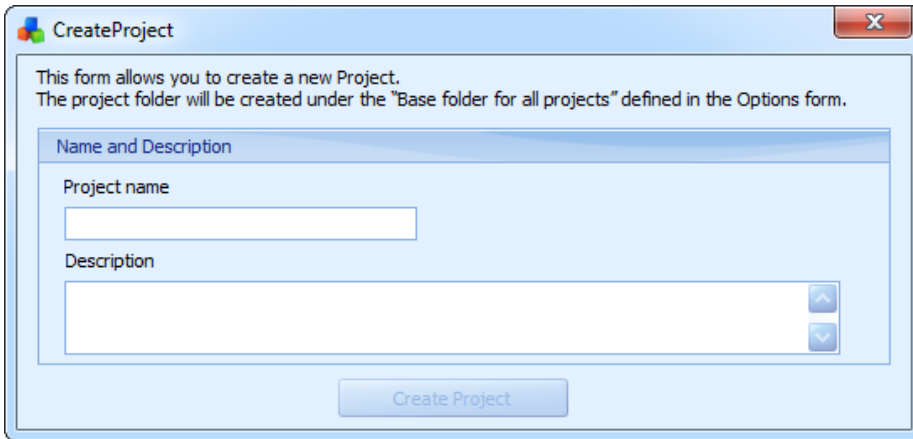
**Contrast (%):** This setting merely establishes the color contrast level used for the above referenced screen check.

## CREATE A PROJECT

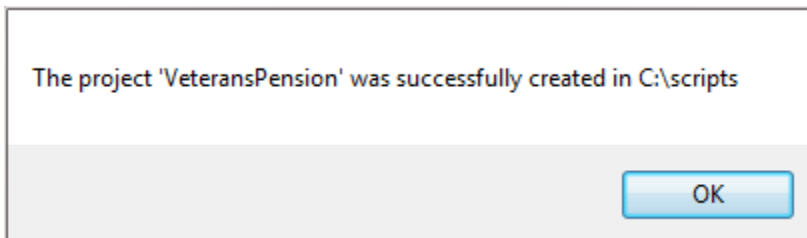
To create a Project, from the “File” menu, click “Project Manager” . The “Project Manager” window opens;



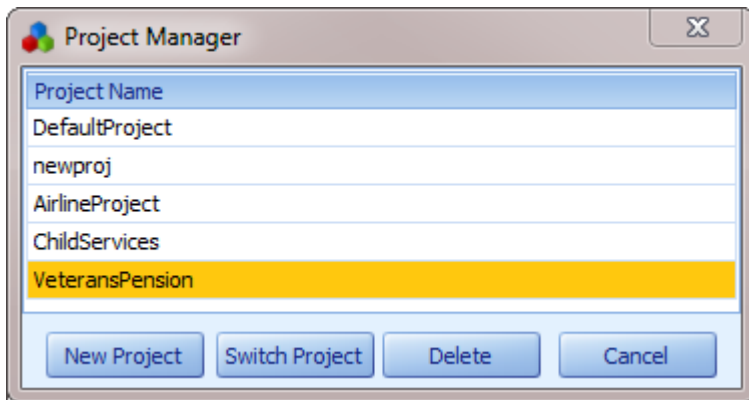
Click the "New Project" button. The "Create Project" window opens;



Provide a name for the Project and click "Create Project". ScenarioBuilder acknowledges the creation of the Project;



A Project folder will be created, with subfolders for Images, Variables, Scenario XML files, Components and documents. ScenarioBuilder automatically makes the new Project the "Active Project";

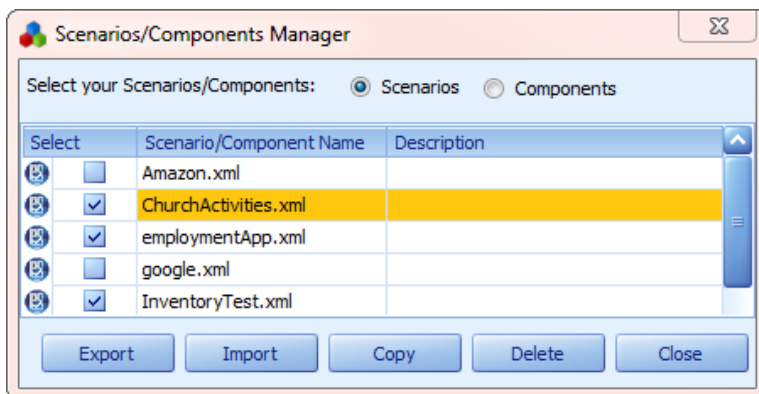


## IMPORT AND EXPORT SCENARIOS

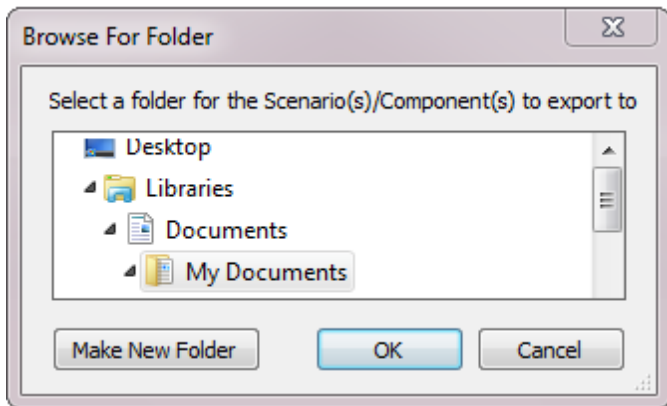
Use “Import” and “Export” to move scenarios in and out of Projects residing in different work spaces. All the sub-content such as Images, Variables, and Components will accompany imported and exported scenarios.

### EXPORT SCENARIO

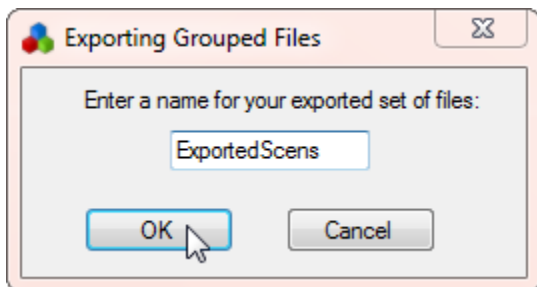
In the “Scenarios/Components Manager” window click the “Scenarios” radio button. Select the scenario(s) to export and click the “Export” button;



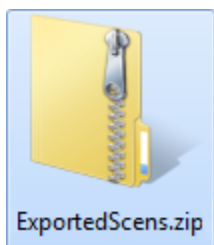
Navigate to an existing directory or create a new folder for your export destination. Note that the export folder is a neutral location from which the Scenario(s) will be imported; it is not intended to be the final location (i.e. the Project folder) for the Scenario(s);



Click “OK”. The “Exporting Grouped Files” window pops up. Enter a name for the exported set of files and click “OK”;



Your exported Scenario(s) will be saved as a .zip folder;

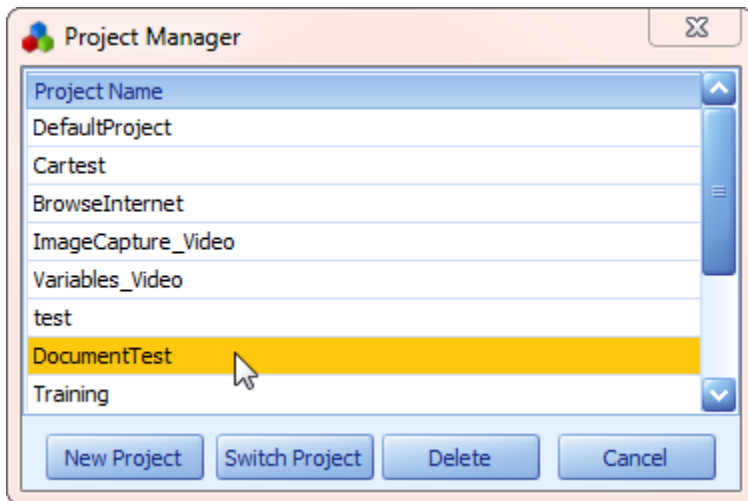


---

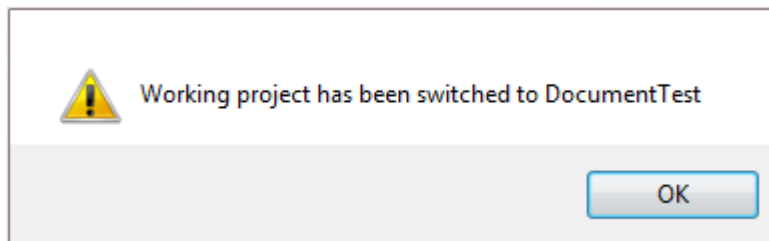
## IMPORT SCENARIO

After exporting your Scenario(s), open the existing Project, or create a new Project into which you wish to import the Scenario(s) (if importing to a Project in a different base folder, change the Project base folder from the “Options” menu);

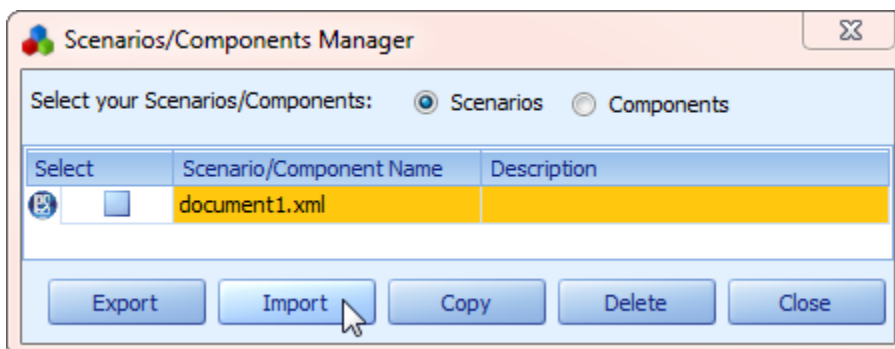
Click the “Project Manager” **Project Manager** option in the “File” menu. The Project Manager opens;



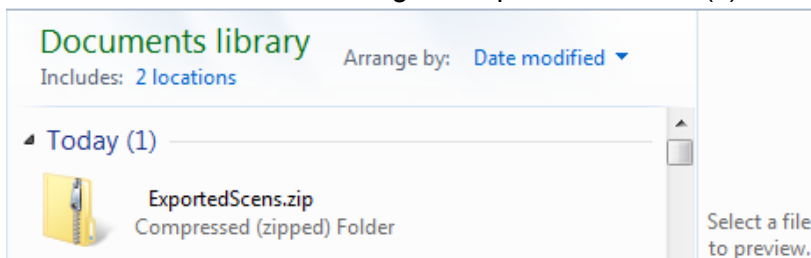
Select the Project to which you want to switch and click “Switch Project” button. ScenarioBuilder will notify you that the Project folder has been switch;



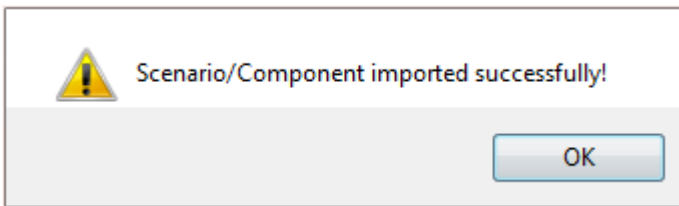
Now that you are working in a new Project folder, you can import the Scenario(s). Open the “Components and Scenarios Manager” and click “Import” button;



Browse to the folder containing the exported Scenario(s);



Double-click the desired .zip file. ScenarioBuilder will acknowledge the import;



Imported Scenario(s) will become available in the current Project.

## NESTING SCENARIOS

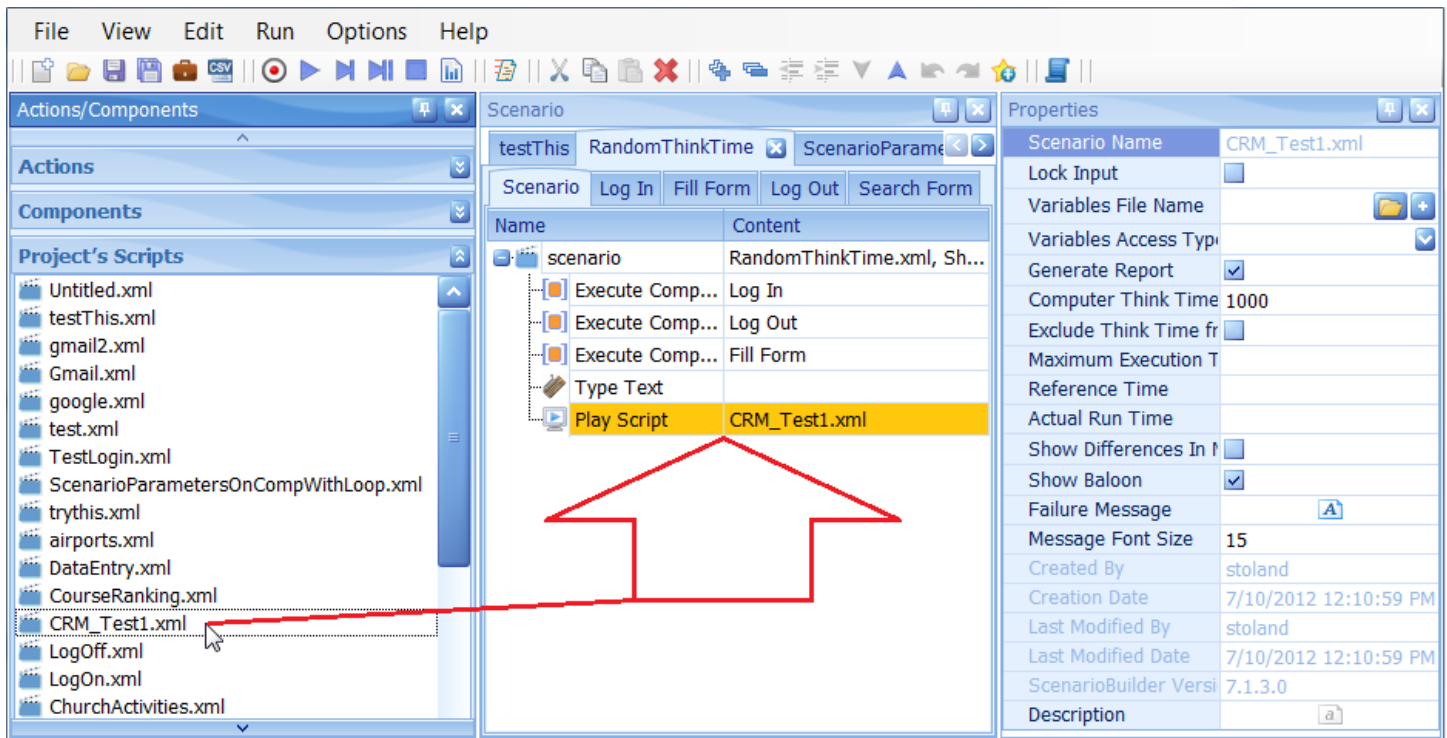
Scenarios can be called by other scenarios. This technique, known as “nesting”, can save you time when developing a set of test scenarios that call the same steps. For example, say you want to create the following set of scenarios to test your CRM application:

- Test 1: Log on to the CRM system, lookup an account, then log off
- Test 2: Log on to the CRM system, lookup a contact, then log off
- Test 3: Log on to the CRM system, lookup a lead, then log off

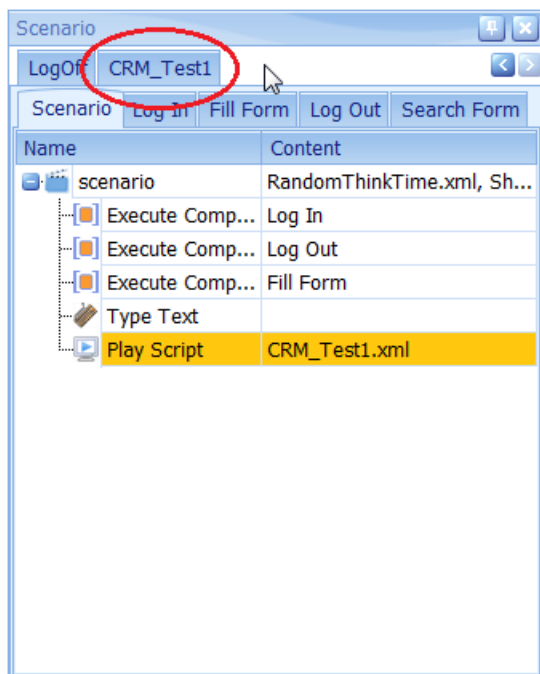
All three scenarios share the “log on” and “log off” sequences. Create one scenario that logs on and another scenario that logs off, then use the “Play Script” Action to call the scenarios into the three required test scenarios;

Scenario	
LookUpAcct	
Name	Content
scenario	CRM_Test1.xml
Play Script	LogOn.xml
Execute Component	LookUpAcct
Play Script	LogOff.xml

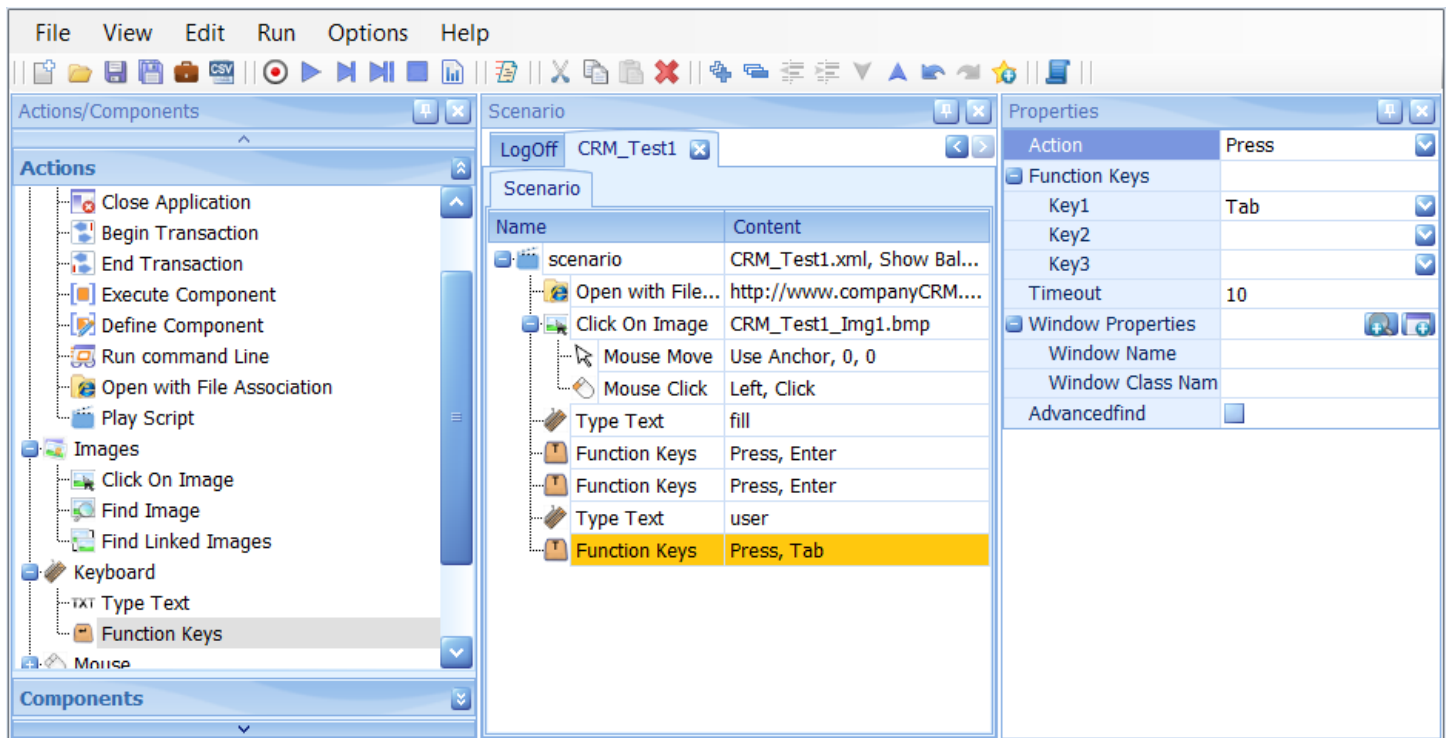
ScenarioBuilder displays all of the scenarios from the current Project in the “Project’s Scripts” pane located in the lower left hand corner. Double-click a script from this pane to add it via the “Play Script” Action;



Notice that a tab for the script has been added above the Scenario Window;



Click the tab to edit the script without having to exit your current scenario;



## WORKFLOWS

Nesting scenarios can also be used to create “Workflows” for Load Tests. A Workflow is simply a Scenario, serving as a scheduler, which plays a set of scenarios in a certain order. A Workflow does not generate an overall report, but it does generate a report for each of the individual scenarios it plays. Workflows can be useful in load testing with AppLoader, as they allow you to create a test plan within the test plan, providing a means to ensure that the same users proceed through a series of processes. For example, if testing the following processes:

- Log on to the application
- Create an order
- Process the created order
- Log off the application

Create a workflow like this;

Scenario	
Name	Content
scenario	Workflow.xml
Play Script	LogIn.xml
Play Script	CreateOrder.xml
Play Script	ProcessOrder.xml
Play Script	LogOut.xml

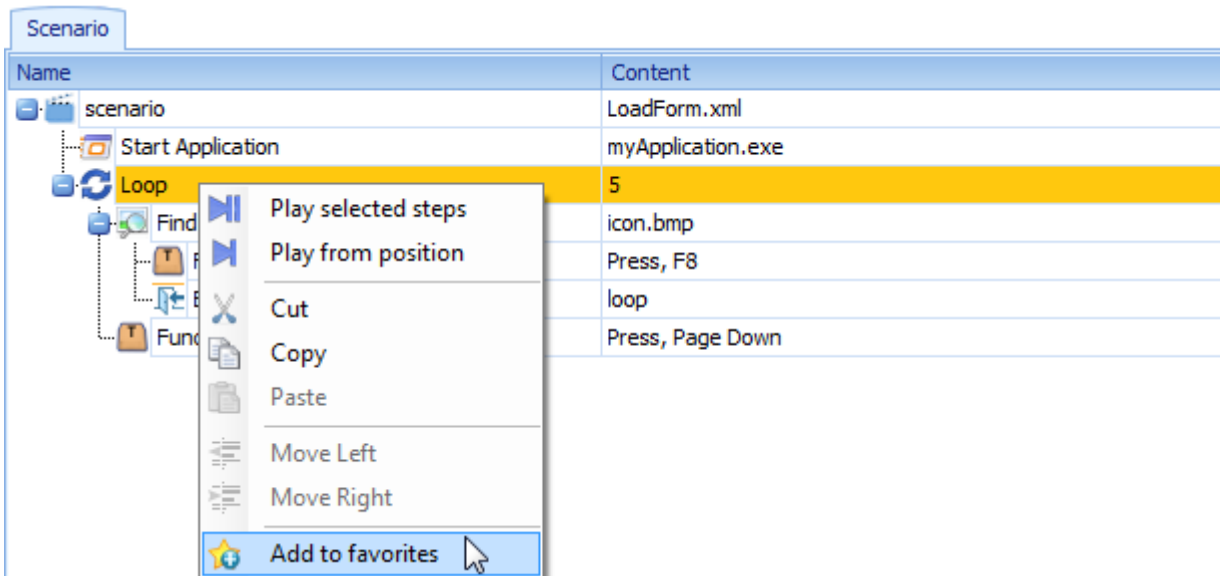
Create multiple workflows to replicate how real users would impact an application. A test plan in AppLoader which uses multiple workflows would look like this;

Details					
Injector	Scenario	rUsers	Delay	Halt	
NRG81	Workflow1.xml	150	5	<input type="checkbox"/>	
NRG81	Workflow2.xml	250	5	<input type="checkbox"/>	
NRG81	Workflow3.xml	400	5	<input type="checkbox"/>	

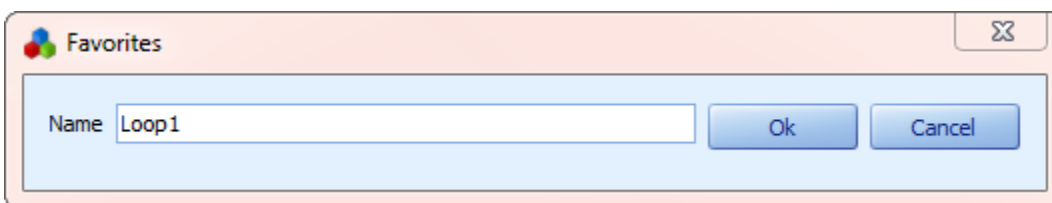
Click here for more about [Using Scenarios with AppLoader](#).

## FAVORITES

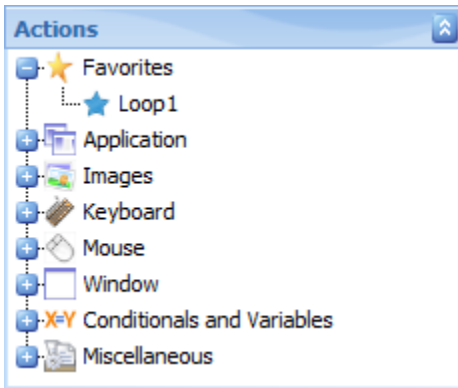
Add an Action to the Favorites folder to be called later, in the current scenario or in a future scenario (from the same Project). Actions saved to Favorites include their child Actions, and they retain their Property settings. Right-click the desired Action (don't highlight the child Actions) and select "Add to favorites" (or highlight the Action and click the "Add to Favorites" tool bar icon);



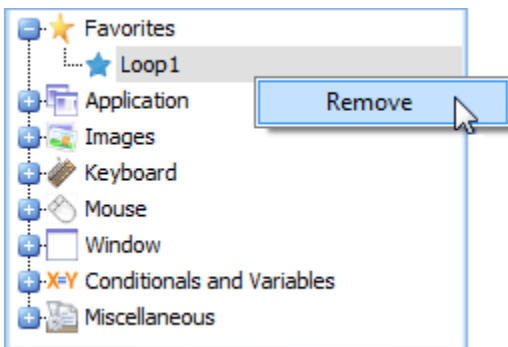
The "Favorites" window opens. Provide a name for the Action and click "Ok";



Favorites can be found in the "Actions" pane of the "Actions/Components" window;

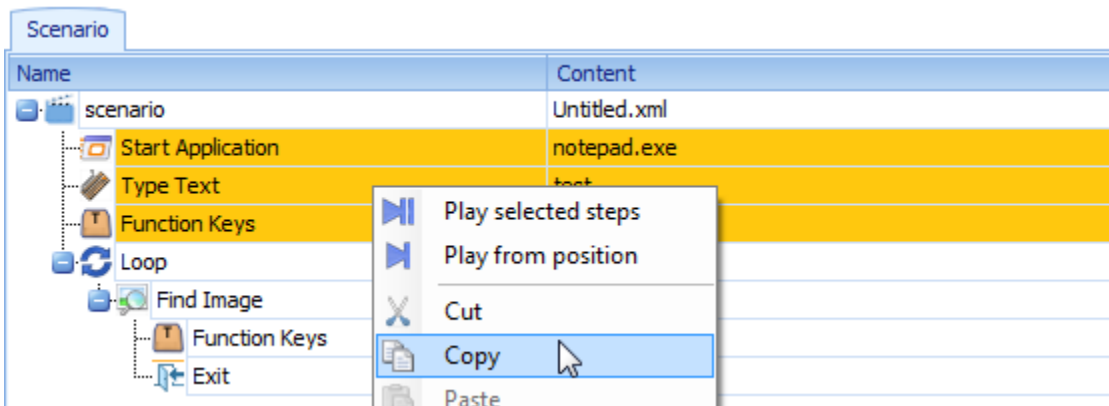


Double-click or drag the “Favorite” item to add it to the Scenario, just as you would with any Action. Delete an item from the Favorites by right-clicking and choosing “Remove”;

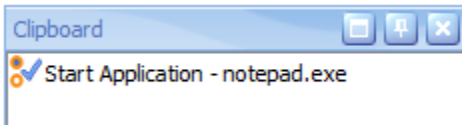


## CLIPBOARD

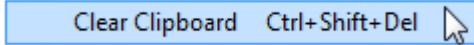
Copy an Action or a series of Actions to the Clipboard to be recalled later in the current scenario. Actions copied to the Clipboard retain their Property settings. Use the Clipboard when Actions with the same properties are repeated in different places within the scenario. Highlight the Action(s) and right-click. Choose “Copy” from the menu;

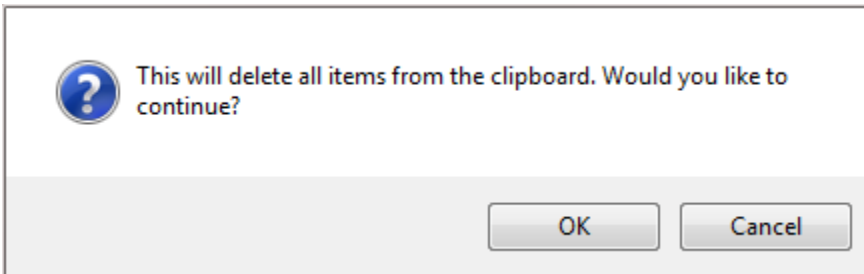


The item(s) will be added to the Clipboard (under the name of the first row of selected Actions);



Double-click or drag the “Clipboard” item to add it to the Scenario, just as you would with any Action.

To clear the Clipboard, select “Clear Clipboard”  from the “Edit” menu;

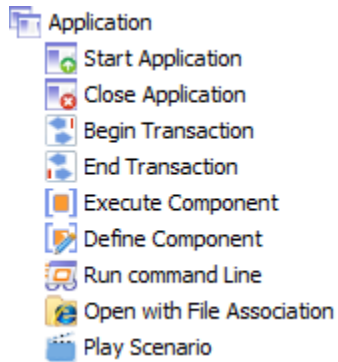


Click “OK” to confirm clear. Note that the Clipboard is cleared whenever ScenarioBuilder is closed.

## ACTION GLOSSARY

This section defines ScenarioBuilder's Actions and their associated properties. Use this guide as a reference to the use and configuration of Actions.

### APPLICATION



#### START APPLICATION

Launch a program via its executable file.

**Application Name:** To launch an executable that is in the Windows path, simply enter the executable name (e.g. to open the *Notepad* application, simply enter *notepad.exe*). If the executable is *not* in the Windows path, provide the full path of the executable (e.g. *c:\program files\internet explorer\explore.exe*). It is acceptable to add parameters to the executable (e.g. *c:\Program Files\Internet Explorer\explore.exe http://www.google.com*).

**Working Folder** (Optional): Provide the application's working folder.

**Continue on Failure:** If enabled, the execution of the scenario will continue if this step fails.

#### CLOSE APPLICATION

**Application Name:** Enter the name of the open program you wish to close. Do not use the full path to the executable.

**Application Started by this Scenario:** If enabled, closes only the instances of this application that were launched by the "Start Application" Action from this scenario.

**Use Task Kill:** If enabled, ScenarioBuilder uses "TaskKill" command to close application by ending its process (unsaved data is lost).

**Continue on Failure:** If enabled, the execution of the scenario will continue if this step fails.

---

## BEGIN TRANSACTION

Use Transactions to mark specific steps in a scenario for response time tracking. For example, a Transaction can be used to measure the time it takes for a user to open and log into an application. Add “Begin Transaction” right *before* the step where you’d like the measurement to begin.

**Transaction Name:** Specify a meaningful name for the Transaction. This is how the Transaction will be identified on reports and graphs

**Reference Time (seconds):** The response time standard to which all subsequent iterations of the Transaction are compared.

---

## END TRANSACTION

Use Transactions to mark specific steps in a scenario for response time tracking. For example, a Transaction can be used to measure the time it takes for a user to open and log into an application. Add **End Transaction** *after* the last step of the sequence you’re measuring.

**Transaction Name:** Use the drop-down list to retrieve the Transaction that you want to end.

**Reference Time (seconds):** The response time standard to which all subsequent iterations of the Transaction are compared. Value is set automatically when “Set Reference Time” button is clicked on “Execution Report”.

---

## EXECUTE COMPONENT

Use to call and execute a defined Component. Refer to [Define Component](#) for more information on defining a Component.

**Component Name:** Use the drop-down menu to specify the block you wish to execute. You should define your Component first.

**Reg Expression:** Use a regular expression to read the execution results. For example, for ping [www.appswatch.com](http://www.appswatch.com), expression #output# will place what the command returns in the variable output, and expression sent = #sent# will place what is between =space and comma in the SENT variable.

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible values: success or fail. For example, name your variable ExecMyCmdStatus.

**Continue on Failure:** If enabled, the execution of the scenario will continue if this step fails.

## DEFINE COMPONENT

A Component is a container within a scenario consisting of a single Action or a series of Actions. Components can be called more than once in a scenario or at any point in the scenario by using the Execute Component Action. Components break up your scenario into meaningful segments for better script organization and maintenance.

**Name:** Specify a meaningful name for the Component.

**Variables File Name** (Optional): Browse for the CSV file used by this Component. This should be used to attach a user-defined Scenario Parameters file on the Component level (values are only retrievable from within the Component).

**Variables Access Type** (Optional): If Scenario Parameters are used, select if the variables should be accessed sequentially or randomly from file.

**Reference Time (seconds):** The response time standard to which all subsequent iterations of the Component are compared. Value is set automatically when “Set Reference Time” button is clicked on “Execution Report” (if “Report” property is enabled).

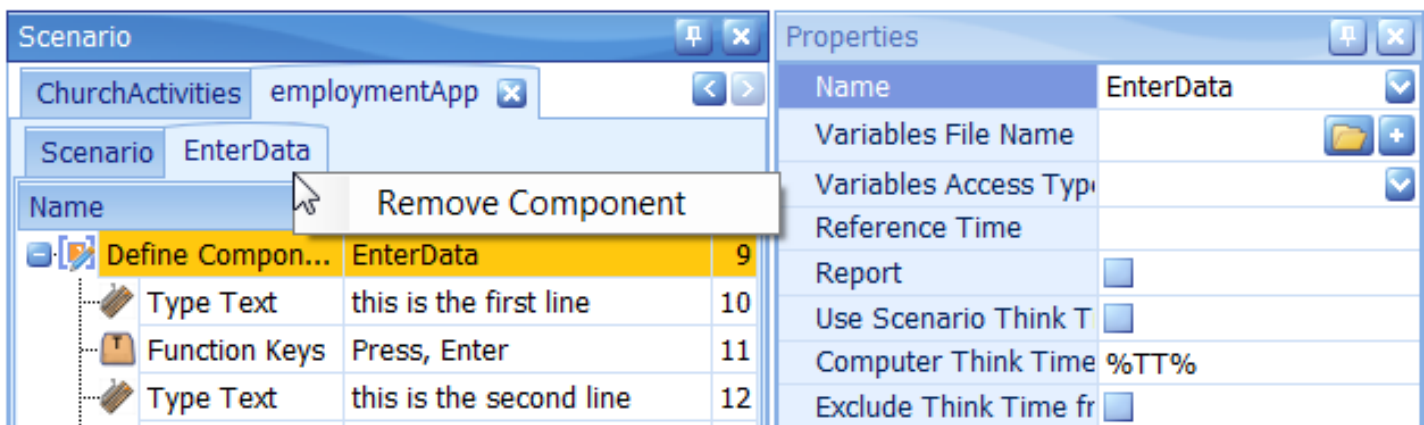
**Report:** If enabled, execution time of the Component will be included in reports and graphs.

**Use Scenario Think Time:** If enabled, this Component will use the same “Think Time” as that which has been established in the scenario properties.

**Computer Think Time for Component:** If not using scenario Think Time, set the Think Time for the Component (value in milliseconds).

**Exclude Think Time from Report:** If enabled, the Think Time will not be included in Component’s execution time. (Note: Setting is only relevant if “Report” property is enabled).

*To remove a Component from your Scenario, simply right-click on the tab and select **Remove Component**.*



---

## RUN COMMAND LINE

Use to execute a program/command at the Operating System level. For example, ping [www.AppsWatch.com](http://www.AppsWatch.com). This Action has the same functionality as Start -> Run from the Windows menu bar.

**Program/Command Name:** Enter the command such as iexplore.exe or launch any file such as c:\Program Files\AppLoader\Agent\ALStation.exe

**Working Folder (Optional):** Enter the working folder.

**Timeout (seconds):** Define the time allotted for the command to return results. If the Action does not complete within this specified time, the step finishes with a FAIL status.

**Reg Expression:** Use a regular expression to read the execution results. For example, for ping [www.appswatch.com](http://www.appswatch.com), expression #output# will place what the command returns in the variable output, and expression sent = #sent# will place what is between =space and comma in the SENT variable.

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible results: *success* or *failure*.

**Continue on Failure:** If enabled, the execution of the scenario will continue if this step fails.

---

## OPEN WITH FILE ASSOCIATION

Use to launch a file or an application with the program associated with it. For example, if you want to launch a URL, this Action will open the URL in the default Internet browser.

**Open Associated Name:** Type the name of the file/application to be opened. For example, for a website address, type the full path: <http://www.AppsWatch.com>. Make sure to include the http:// and www as needed. The scenario will then attempt to open this address in the default browser.

**Working Folder (Optional):** Enter the working folder.

**Timeout (seconds):** Define the time allotted for the application/file to launch. If the application does not launch within this specified time, the step finishes with a FAIL status.

**Reg Expression:** Use a regular expression to read the execution results. For example, for ping [www.appswatch.com](http://www.appswatch.com), expression #output# will place what the command returns in the variable output, and expression sent = #sent# will place what is between =space and comma in the SENT variable.

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible results: *success* or *failure*.

**Continue on Failure:** If enabled, the execution of the scenario will continue if this step fails.

## PLAY SCENARIO

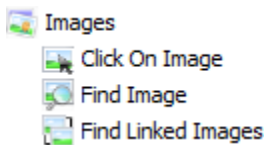
This feature allows you to nest an unlimited number of scenarios.

**Scenario Name:** Browse for a saved scenario and select the scenario file you wish to include (file extension for saved scenarios are .xml).

**Continueonfail:** Check this field if you would like your main scenario to continue even if this selected (or nested) scenario returns as a FAIL.

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible values: *success* or *failure*.

## IMAGES



## CLICK ON IMAGE

With this Action, ScenarioBuilder looks for and then clicks on an image. The “Click on Image” Action brings with it two *child* Actions into the Scenario Window ([Mouse Move](#); [Mouse Click](#)).

**Images:** If you wish to add more than one image to the Action, simply click on the + icon to the right of Images property field. You then can select your Image #2, #3, and so forth. If you have two or more images, the Action will pass if any one of the specified images is visible on the screen. For example, if you selected bitmap1 and bitmap2, then the scenario will look for both bitmaps and it will proceed to the next Action once it has found the first one.

**Find Image #:** Right click in this field to determine how you want to select the image:

**Capture Image:** Select if you need to take a screenshot of an image. Your cursor will change to a screen capture tool. Drag the cursor around the image you wish to take and provide a meaningful name for it.

**Load:** Select if you have previously captured an image and wish to use it for this Action.

**Contrast Level** (can only be used after an image has been “captured” or “loaded”): This will launch the Black and White Contrast Display tool which allows you to test and set a contrast level to save the image in black and white. Use this feature if variations in color are preventing ScenarioBuilder from consistently recognizing and finding an image.

**View Image** (can only be used after an image has been “captured” or “loaded”): This will open the image in your system’s default viewer.

**Remove Image:** Select if you wish to remove the image from this Action.

**Timeout (seconds):** Define the time allotted for the image to be found. If the image is not found within this specified time, the step finishes with a FAIL status.

**Continue on Failure:** Enable this property if you would like your scenario to continue even if the bitmap was not found.


**Tolerance:** Set the color matching tolerance percentage. The default value (0) matches the exact colors and finds images faster. Note: use this feature especially with Flash and Flex applications as the colors vary with each rendering on a High Color (16 bit) display. Warning: a tolerance higher than 50% may cause false positives.

**Black and White Contrast Ratio:** A value greater than 0 causes ScenarioBuilder to search for image in black and white. Use the [Black and White Contrast Level](#) tool to find an appropriate setting.

**Anchor Position X:** Specifies where a mouse click will occur horizontally on the image (the “x” coordinate). The default setting is “Center”.

**Anchor Position Y:** Specifies where a mouse click will occur vertically on the image (the “y” coordinate). The default setting is “Center”.

**Sleep Time (milliseconds):** This is the amount of time scenario sleeps between image checks. For example, if “Timeout” is set to 10 seconds and “Sleep Time” is set to 500 milliseconds, the scenario will check for image a total of 20 times.

**Image Search Area:** Click the “+”  icon and use the tool to outline an area in which scenario will search for image. Be sure the search area covers the entire image you've taken. Use this feature only when the Image stays on a fixed location on the screen. If specified, the scenario will look for the image in the specified location; otherwise, the scenario will search the entire desktop for the image. In both cases, the scenario searches within milliseconds; however, when the search area is specified, the search is even faster.

**Wait for Image to Disappear:** If enabled, Action name (in Scenario Window) changes to “Wait Image Disappear”. Step will succeed if image is not present. Usually this Action is used immediately after a “Click on Image” Action that clicks the same image, to ensure that the application has responded to the click (opened the new page; advanced to the link; etc.).

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible values: *success* or *failure*.

---

## FIND IMAGE

Scenario looks for an image just as an end-user would. This Action can be used to verify that a step has occurred properly (for example, finding a logo, image or banner to confirm that a web page has loaded, before typing text). The “Find Image” Action properties are identical to the “Click on Image” properties (the only

difference in the two Actions being that “Click on Image” includes the “Mouse Move” and Mouse Click” child Actions). Please see [Click on Image](#) section for properties review.

## FIND LINKED IMAGES

Find images in a grid to identify and select a specific row or record.

**Images:** Click on the camera icon at right to capture an image using ScenarioBuilder’s tool. Your cursor will change to a screen capture tool. Drag the cursor around the image you wish to take and provide a meaningful name for it. Repeat the process, taking enough images from a row to uniquely identify it.

**Timeout (seconds):** Define the time allotted for the images to be found. If the images are not found within this specified time, the step finishes with a FAIL status.

**Tolerance:** Set the color matching tolerance percentage. The default value (0) matches the exact colors and finds images faster. Note: use this feature especially with Flash and Flex applications as the colors vary with each rendering on a High Color (16 bit) display. Warning: a tolerance higher than 50% may cause false positives.

**Black and White Contrast Ratio:** A value greater than 0 causes ScenarioBuilder to search for image in black and white. Use the [Black and White Contrast Level](#) tool to find an appropriate setting.

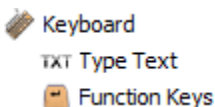
**Anchor Position X:** Specifies where a mouse click will occur horizontally on the image (the “x” coordinate). The default setting is “Center”.

**Anchor Position Y:** Specifies where a mouse click will occur vertically on the image (the “y” coordinate). The default setting is “Center”.

**Sleep Time (milliseconds):** This is the amount of time scenario sleeps between image checks. For example, if “Timeout” is set to 10 seconds and “Sleep Time” is set to 500 milliseconds, the scenario will check for image a total of 20 times.

**Execution Status Variable (optional):** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of this execution. This variable will have two possible values: *success* or *failure*.

## KEYBOARD




## TYPE TEXT

Use Type Text when you need to enter a single character or string of text.

**Text:** Click the Text field to enter the text you wish to be typed then click “OK”.

**Encrypted Text:** Check this box if you wish to encrypt the text for security purposes. This is used mainly for passwords.

**Windows Properties (optional):** You can specify the window where you wish to type the function key entries. To select the window: click on the  **Search**, highlight the desired window with your cursor and hit the **CTRL** key. After you have selected a window, the **Window Name** and **Window Class Name** fields will automatically be populated. Check these names to make sure you have selected the correct window.

**Timeout (seconds):** Define the time allotted for the image to be found. If the image is not found within this specified time, the step finishes with a FAIL status.

**Advancedfind:** When checked, the scenario will try to find the best match down to the first 4 letters of the window's name and class.

**Delay Key Strokes (milliseconds):** Specify the delay between each keystroke.

---


## FUNCTION KEYS

Use Function Keys when you need to press/hold function keys, such as Enter, F6, Tab, etc. Also can be used when a combination of keys must be struck at once.

**Action:** Select the keyboard action you wish to execute from the drop down box (*Combination, Down, Press, Up*).





**Function Keys:** Select the function key(s) you wish to type. Use the drop-down box to select each function key. For example, you could set Function Key 1: Ctrl, Function Key 2: Alt, and Function Key 3: Shift.

**Timeout (seconds):** Define the time allotted for the image to be found. If the image is not found within this specified time, the step finishes with a FAIL status.

**Windows Properties (optional):** You can specify the window where you wish to type the function key entries. To select the window: click on the  **Search**, highlight the desired window with your cursor and hit the **CTRL** key. After you have selected a window, the **Window Name** and **Window Class Name** fields will automatically be populated. Check these names to make sure you have selected the correct window.

**Advancedfind:** When checked, the scenario will try to find the best match down to the first 4 letters of the window's name and class.

## MOUSE

-  Mouse
-  Mouse Click
-  Mouse Move and Click
-  Mouse Move

---

## MOUSE CLICK

Execute a mouse click. Use to select an icon in an application or a product photo on a web page.

**Button:** Use the pull-down menu to select whether the scenario should click the right, left, or middle button on the mouse.


**Click Type:** Use the pull-down menu to define the click type (*Click, Double Click, Up, or Down*).

---

## MOUSE MOVE AND CLICK

This Action combines the mouse “Move” and “Click” properties into a single Action.

**Use Anchor:** If enabled, mouse location is relative to the parent *Image* or *Window* Action (property is only relevant if the “Mouse Move” Action is a child of either an Image or Window Action)

**Mouse Coordinates:** Use the “Mouse Coordinates”  tool to click anywhere on the desktop and automatically set the x and y coordinates.

**x:** Manually enter a value for the “x” coordinate for the mouse click (if tool is used, this value is automatically populated).

**y:** Manually enter a value for the “y” coordinate for the mouse click (if tool is used, this value is automatically populated).

**Button:** Use the pull-down menu to select whether the scenario should click the right, left, or middle button on the mouse.


**Click Type:** Use the pull-down menu to define the click type (*Click, Double Click, Up, or Down*).

---

## MOUSE MOVE

Specify mouse (cursor) location.

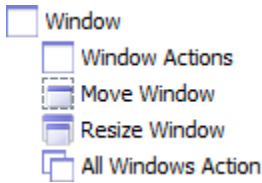
**Use Anchor:** If enabled, mouse location is relative to the parent *Image* or *Window* Action (property is only relevant if the “Mouse Move” Action is a child of either an Image or Window Action)

**Mouse Coordinates:** Use the “Mouse Coordinates”  tool to click anywhere on the desktop and automatically set the x and y coordinates.

**x:** Manually enter a value for the “x” coordinate for the mouse click (if tool is used, this value is automatically populated).

**y:** Manually enter a value for the “y” coordinate for the mouse click (if tool is used, this value is automatically populated).

## WINDOW





---

## WINDOW ACTIONS

Use to manipulate the behavior of windows on the screen.

**Window Action Name:** Select the desired window Action (*Close, Focus, Maximize, Minimize, Restore, or Waitwindow*).

**Windows Properties:** Specify the window for this Action. To select the window, click the “search”  icon, highlight the desired window with your cursor and hit the CTRL key. After you have selected a window, the **Window Name** and **Window Class Name** fields will automatically be populated. Check these names to make sure you have selected the correct window.

**Advancedfind:** When checked, the scenario will try to find the best match down to the first 4 letters of the window's name and class.

**Timeout (seconds):** Define the time allotted for the window to be found. If the window is not found within this specified time, the step finishes with a FAIL status.

**Sleep Time (milliseconds):** This is the amount of time scenario sleeps between window checks. For example, if “Timeout” is set to 10 seconds and “Sleep Time” is set to 500 milliseconds, the scenario will check for the window a total of 20 times.

**Wait for Window to Disappear:** Enable this property if you want the scenario to wait for the window to disappear from the screen.

**Continue on Failure:** Enable this property if you would like your scenario to continue even if the window Action is not completed successfully.

**Execution Status Variable:** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of its execution. This variable will have two possible results: *success* or *failure*.


**Anchor Position X:** This is the x coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.

**Anchor Position Y:** This is the y coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.

---

## MOVE WINDOW

Use this Action to move a window to a new location on the screen.

**Windows Properties:** Specify the window for this Action. To select the window, click the “search”  icon, highlight the desired window with your cursor and hit the CTRL key. After you have selected a window, the **Window Name** and **Window Class Name** fields will automatically be populated. Check these names to make sure you have selected the correct window.

**Advancedfind:** When checked, the scenario will try to find the best match down to the first 4 letters of the window's name and class.

**Timeout (seconds):** Define the time allotted for the window to be found. If the window is not found within this specified time, the step finishes with a FAIL status.

**Sleep Time (milliseconds):** This is the amount of time scenario sleeps between window checks. For example, if “Timeout” is set to 10 seconds and “Sleep Time” is set to 500 milliseconds, the scenario will check for the window a total of 20 times.


**Wait for Window to Disappear:** Enable this property if you want the scenario to wait for the window to disappear from the screen.

**Continue on Failure:** Enable this property if you would like your scenario to continue even if the window Action is not completed successfully.

**Execution Status Variable:** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of its execution. This variable will have two possible results: *success* or *failure*.

**Anchor Position X:** This is the x coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.


**Anchor Position Y:** This is the y coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.

**Coordinates:** Enter the desired destination coordinates on the screen. Click on the  tool at right. The mouse cursor will become a ‘+’ sign and you may click a different area while using the captured image as a reference point.

---

## RESIZE WINDOW

This Action modifies the window to the desired width and height.

**Windows Properties:** Specify the window for this Action. To select the window, click the “search”  icon, highlight the desired window with your cursor and hit the CTRL key. After you have selected a window, the **Window Name** and **Window Class Name** fields will automatically be populated. Check these names to make sure you have selected the correct window.

**Advancedfind:** When checked, the scenario will try to find the best match down to the first 4 letters of the window's name and class.

**Timeout (seconds):** Define the time allotted for the window to be found. If the window is not found within this specified time, the step finishes with a FAIL status.

**Sleep Time (milliseconds):** This is the amount of time scenario sleeps between window checks. For example, if “Timeout” is set to 10 seconds and “Sleep Time” is set to 500 milliseconds, the scenario will check for the window a total of 20 times.

**Wait for Window to Disappear:** Enable this property if you want the scenario to wait for the window to disappear from the screen.

**Continue on Failure:** Enable this property if you would like your scenario to continue even if the window Action is not completed successfully.

**Execution Status Variable:** Assign a property variable to this Action if you will later need to take other Actions depending on the success or failure of its execution. This variable will have two possible results: *success* or *failure*.

**Anchor Position X:** This is the x coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.

**Anchor Position Y:** This is the y coordinate. Specifies where the mouse will move if a move mouse Action follows. The position is Center by default.

**Size:** Enter the desired height and width for the window.

---

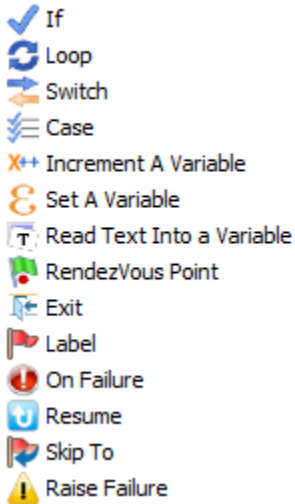
## ALL WINDOWS ACTION

Use when you want to close or minimize all open windows on the desktop.

**Action:** Select the window Action to be taken (*close* or *minimize*).

## CONDITIONALS AND VARIABLES

## X=Y Conditionals and Variables



## IF

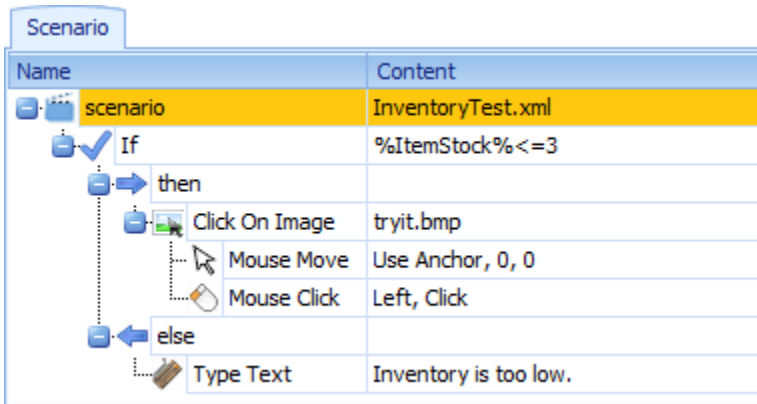
An “If” Action allows you to set up an “If-Then-Else” condition in your scenario that will execute one of two courses of action, depending on whether the condition is a success (*True*), or a failure (*False*).

Expression: The condition to evaluate as being either True or False (e.g. %Variable%==1; %ReturnedValue%!=Error; %ExecMyImageStatus%==success).

Use the following comparison operators in your expressions:

Operator	Meaning
==	Equal
!=	Not Equal
>	Greater Than
>=	Greater Than or Equal
<	Less Than
<=	Less Than or Equal

Next, below the “Then” row in the Scenario Window, add the Action(s) that should occur when the condition is true. Last, below the “Else” row in the Scenario Window, add the Actions(s) that should occur when the condition is false. Be sure that the Actions below the “Then” and “Else” rows are indented, making them child Actions. Below is an example of an “If” statement:



For more about working with variables in ScenarioBuilder, please refer to Scenario Parameters in this guide.

## LOOP

Use to repeat an Action or series of Actions a specified number of times.

**Iterations:** Specify the number of times the Actions in the Loop should be repeated.

Drag Actions into the Scenario Window under the Loop Action. Be sure they are indented, making them child Actions of the Loop.

## SWITCH

Use Switch when you have different values for a variable and you need to apply different Actions based on the value retrieved. Switch/Case is very similar to the IF/THEN/ELSE. It is normally used when you want to find an equal result, and not a greater or less than result.

**Variable Name:** After adding **Switch** to your Scenario, enter a Variable Name. Be sure to surround the variable by percentage symbols (e.g. %variable1%).

Next, you will need to add a Case to your Switch sequence. See the description of Case below.

## CASE

Used with the Switch Action.

Once you have added **Switch** to your scenario and defined the variable, then you can drag the **Case** Action over to your Switch sequence. For each **Case**, define the variable value that will make the case succeed and execute its Action.

## INCREMENT A VARIABLE

Use the Increment A Variable Action when you want to increment a set variable.

**Variable Name:** Enter the name of the variable you wish to increment (the “Set A Variable” Action must precede this step, establishing the variable to be incremented).

**Increment Value:** Enter the value that you wish to increment the entered variable by.

---

## SET A VARIABLE

Use to add a pre-defined variable to your scenario.

**Variable Name:** Give a meaningful name to your variable (e.g. *loginnames* or *ImageFound*)

*Note: The Variable Name should not include any special characters or spaces*

**Variable Value:** Enter your variable value as follows:

Pre-defined variable: simply type in your value or text entry

Pre-defined variable: use the pull-down menu at right to select your pre-defined variable. Options include a 4-digit random number, or a date in the format mm/d/yyyy or mm/dd/yyyy

For date and time pre-defined variable values, the default values are always based on current date and time. However, you can add or subtract values from the individual month, date, year, hour, minute or second. For example, [M+1/d/yy], [MM/dd-1/yy], [h-1:mm:ssstt].

*Note: Do not remove the beginning and ending brackets from the pre-defined values; only one modification (addition or subtraction) is allowed per variable value.*

---

## READ TEXT INTO A VARIABLE

This Action copies selected formatted text, or translates selected characters from an image (using optical character recognition (OCR) technology) into text, then reads the value into a variable.

**Variable Name:** Give a meaningful name to your variable (i.e. a name which represents the characters within the image).

**Copy/OCR:** Select “Copy” when capturing formatted text; select OCR when capturing an image of text.

**Use Anchor:** If checked, the relative coordinates to the parent image or window will be used.

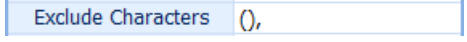
**OCR Type:** Select the type which best describes the nature of the text being captured. Recommended settings are “Single Text Line” or “Single Word”.


**Language:** Select the language in which the text is written to assist OCR in converting to actual text.

**Scale Factor:** The scale factor is used to blow up an image by this factor to properly recognize the image. The **Scale Factor** is set to 2 by default.

**Contrast:** Adjust the contrast to optimize OCR. The default value is 75.

**Include Characters:** If text is all uppercase, enter “A” in this field; if all lowercase, enter “a” in this field. If mixed, leave this field empty.

**Exclude Characters:** Use when certain characters should be omitted from the captured text. For example, if capturing a phone number that is displayed (888)624-4447, excluding parenthesis and dashes will result in the value being read as: 8886244447. Note: if excluding more than one character, list in this property field *without* any separators, e.g. .

**Text Area:** Click the ellipses  to use your mouse to select the area from which to capture text. Capture only the desired text. Capturing extraneous data, even pixels from surrounding area, can cause OCR to fail. Note: the “Start” and “End” Text Area “X” and “Y” property fields will automatically populate after you make your selection.

---

## RENDEZVOUS POINT

Rendezvous points instruct multiple rUsers to wait during the test execution in order to simultaneously perform a task. For example, to emulate peak load on a Citrix server, you can insert a rendezvous point instructing 100 rUsers to click on the update button at the same time.

Put the **Rendezvous Point** right before the scenario step where you want peak load (all users). Give the **Rendezvous Point** a meaningful name in the properties pane.

*Note: Rendezvous Point is only applicable to AppLoader; this Action will be ignored in AppsWatch.*

---

## EXIT

Exit stops the execution of a Loop, Component, or scenario. After adding **EXIT** to your Scenario, use the pull-down menu in the right pane to select which section you want to stop: Loop, Component, or scenario. This can be used in Conditional statements; for example, you can loop a Find Image Action five times; if the image is found, you can page down; if the image is not found, you can exit the loop and continue with the scenario.

---

## LABEL

Normally ScenarioBuilder runs the Actions sequentially, but if you wish to skip some Actions based on conditions you can use a Label to do so. A Label is just a name that separates the different Actions. Everything following the Label is considered under that Label.

The scenario is able to jump to labeled Actions by calling their Label name using “Skip To” [label name] Action. The only limitation is that you can’t skip to a label which had been defined in previous steps. The label must be defined afterwards, since “skip to” just looks forward to find the label.

**Label:** Give a meaningful name to your label.

---

## ON FAILURE

On Failure lets you determine what Action(s) to take when a scenario or Component fails. For example, if your Component fails, you might choose to close the application or web page. After adding On Failure to your Scenario, you can drag any Action (i.e. Exit or Close Window) from the Actions panel. These will be the steps taken upon failure.

---

## RESUME

Resume can be used only under an “On Failure” Action. The “Resume” Action allows the scenario to try and continue running from the point where the failure occurred. Use “Resume” to handle failures caused by pop-ups, for example. Follow an Action that closes the pop-up window with a “Resume” Action to pick up where the scenario left off before the pop-up disrupted the execution.

**Retries:** The number of times the scenario will attempt to resume after the failure.

---

## SKIP TO

This Action lets the scenario skip steps and jump to a certain point based on the label.

**Label:** Select the label name that you want the scenario to skip to.

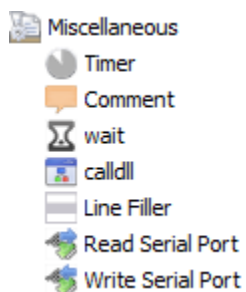
---

## RAISE FAILURE

The Raise Failure Action reports the failure in your test results. If Raise Failure is omitted, the scenario will still return a success status.

---

## MISCELLANEOUS



---

## TIMER

This feature allows you to stop and resume the execution timer. It can be very useful when you want to perform certain Actions (such as cleaning up an environment), which you do not wish to be reported in the overall execution time.

Use the pull-down menu in the Properties pane to select “Suspend” or “Resume” Timer. The time spent between the Suspend Timer and Resume Timer steps will not be included in the reported execution time.

---

## COMMENT

Use this Action to add a descriptive comment as needed. The Comment Action has no effect on the scenario itself; rather, it can be used as a helpful note or explanation you can add for yourself or anyone who might be working with your scenario.

---

## WAIT

Wait periods allow you to add a step in your scenario to give the system enough time to execute the Action, or to load the window/program properly. Use the Wait Action when you need to add more time to execute a line of Action. After adding Wait to your Scenario, enter the wait time (in seconds) in the right pane. The default wait time is 10 seconds.

*Note: make sure the wait time plus the rest of the scenario time does not exceed the playback time.*

---

## CALLDLL

Use CallDll to call a dll function.

**Dllname:** Enter the name of the desired dll; for example, user32.dll. It's important to include the .dll extension.

**Function:** Add the function of the dll.

**Parameter:** Enter the parameter. You can add as many parameters as needed using the + icon.

---

## LINE FILLER

This Action adds a colored line within your scenario. It has no impact on the scenario playback, but is simply used to help you visually organize your scenario. You can modify the line's color by editing the R, G, B values in the Properties window.

---

## READ SERIAL PORT

This Action allows you to read data from ScenarioStation's serial port. After adding Read Serial Port to your Scenario, you will need to configure the settings in the right pane. Select the appropriate port number (i.e. 1 or 2), and then configure the port settings (baud rate, data bits) as needed.

In addition to the port settings, **Reg Expression** uses a regular expression to read the execution results. For example, for ping www.appswatch.com, expression #output# will place what the Action returns in the variable output, and expression sent = #sent# will place the contents between =space and comma in the SENT variable.

## WRITE SERIAL PORT


This Action allows you to write data to a ScenarioStation's serial port.

Select the appropriate port number (i.e. 1 or 2), and then configure the port settings (baud rate, data bits) as needed.

In addition to the port settings, **Reg Expression** uses a regular expression to read the execution results. For example, for ping www.appswatch.com, expression #output# will place what the Action returns in the variable output, and expression sent = #sent# will place the contents between =space and comma in the SENT variable.

## SCENARIO RECORDER GLOSSARY

### CAPTURE IMAGE

The “Capture Image”  Action, when executed from the Scenario Recorder, prompts the “Image Preferences” window after the image is captured;



**Image Display:** Displays the captured image.

**Image Name:** The name your image will be saved as. See Capture Image Preferences to adjust image naming convention.

**Mouse Anchor:** Location of the mouse cursor on the captured image.

**Mouse Position:** Where the cursor will be in reference to the selected image.

**Mouse Action:** The type of click to be performed by the mouser on the image (e.g. *left double click*; *right click*).

**Time Out:** The amount of time the scenario has to find the image before the scenario playback will timeout.



**Accept and Play:** Accepts and plays the step, adding it to the scenario and executing the Action.



**Accept:** Accepts the step, adding it to the scenario without executing the Action.

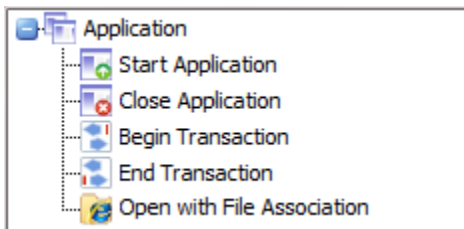


**Cancel:** Discards the Action without adding it to the scenario.

*Push **Esc** to exit the Capture Image mode*

## ACTIONS

The Scenario Recorder offers a limited menu of Actions that can be accessed by clicking the “Actions” button;



[Start Application](#): Launch a program via its executable file.

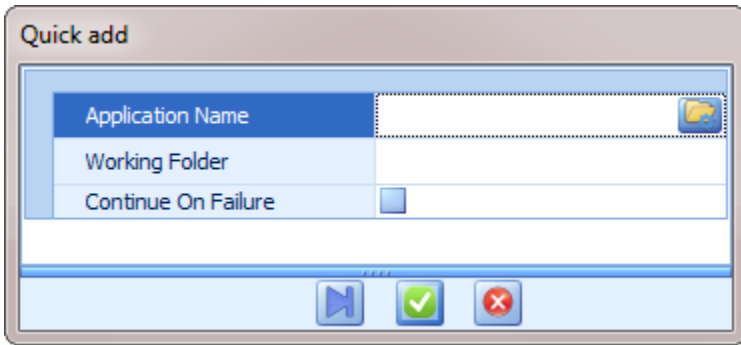
[Close Application](#): Close an open application.

[Begin Transaction](#): Mark the beginning of specific steps in a scenario for response time tracking.

[End Transaction](#): Mark the end of specific steps in a scenario for response time tracking.

[Open with File Association](#): Launch a file or an application with the program associated with it.

Click any of these Actions and the “Quick Add” window opens, in which you may set the Action’s properties (“Start Application” Quick add window shown below);



**Accept and Play:** Accepts and plays the step, adding it to the scenario and executing the Action.



**Accept:** Accepts the step, adding it to the scenario without executing the Action.




**Cancel:** Discards the Action without adding it to the scenario.

## UNDO



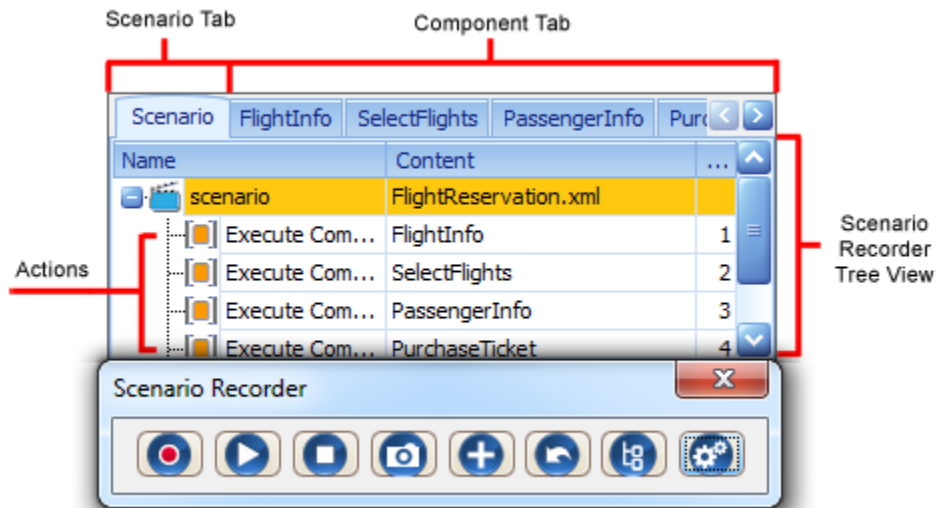
Click “Undo” to reverse the last change made to the Scenario, restoring it to an earlier state. Undo can be used repeatedly to erase a series of changes.

If there are no Actions left, the “Undo” button will gray out .


## TREE VIEW

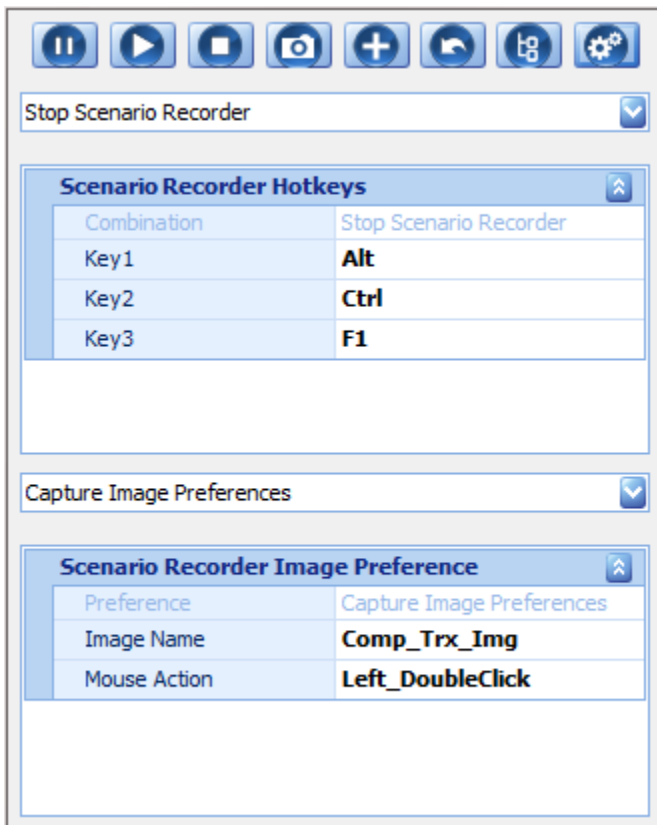


Click the “Tree” View button to hide or display the Scenario Window while remaining in Scenario Recorder mode. Click to display steps as you’re recording your scenario. Switch between different Component tabs while recording to add steps to different Components;

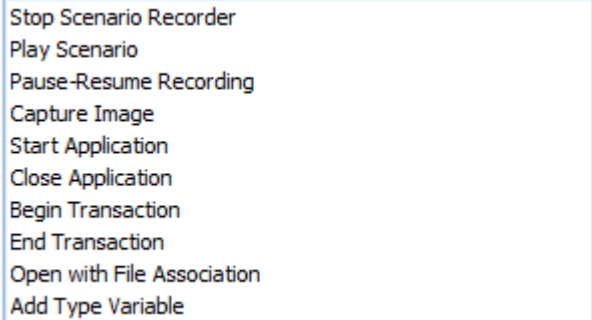


## PREFERENCES

Click the “Preferences”  button to manage hotkeys that trigger Actions or menu options in Scenario Recorder. Also, configure Scenario Recorder “Image” Action *Naming* and *Mouse Action* conventions;



**Scenario Recorder Hotkeys:** Combine up to three keys to create hotkeys that will initiate Actions. Click the arrow in the drop down box and select the Action for which to create a hotkey;



Stop Scenario Recorder  
Play Scenario  
Pause-Resume Recording  
Capture Image  
Start Application  
Close Application  
Begin Transaction  
End Transaction  
Open with File Association  
Add Type Variable

**Stop Recording:** Exit Recorder mode and return to ScenarioBuilder default view.

**Play Scenario:** Play the entire scenario from Scenario Recorder mode.

**Pause-Resume Recording:** Pause or resume a recording session.

**Capture Image:** Initiate the “Capture Image” Action.

**Start Application:** Launch a program via its executable file..

**Close Application:** Close an open application.

**Begin Transaction:** Mark the beginning of specific steps in a scenario for response time tracking.


**End Transaction:** Mark the end of specific steps in a scenario for response time tracking.

**Open File Association:** Launch a file or an application with the program associated with it.

**Add Type Variable** Bring up a list of variables from the CSV file attached to the scenario.

*If you assign a Hotkey combination that is already assigned to another action in Scenario Recorder, a validation error will occur and will reset your hotkey for the selected function. To view the assigned Hotkeys, mouse over the buttons in Scenario Recorder.*

Click the “Preferences”  button again, to close the “Preferences” window and remain in Recorder mode.

Click the “Close”  button in the upper right corner of the “Preferences” window to exit Recorder mode and return to ScenarioBuilder view.

**Scenario Recorder Image Preferences:** Set the captured image naming convention and mouse clicking defaults.

**Image Name:** Set the captured image’s default naming convention. You may change the name of your captured image after you capture it.

*Comp\_Trx\_Img*: Combines the image's Component's name, the Transaction associated with the image, and the image name with an incremented number each time you capture an image. For example, when the Component name is *Phone*, Transaction name is *Call*, your captured image will be saved as *Phone\_Call\_Img1.bmp* and the next image will be saved as *Phone\_Call\_Img2.bmp*, *Phone\_Call\_Img3.bmp*, etc.

*Trx\_Img*: Combines the Transaction associated with image and the image name with an incremented number each time you capture an image.

*Img*: Combines the image name with an incremented number each time you capture an image.

*None*: Does not specify a naming convention. You must decide a name for your captured image each time an image is captured.

**Mouse Action**: Set the mouse click default (e.g. *Left\_Click*-left mouse, single click; *Left\_DoubleClick*-left mouse, double click). You may change this setting after you capture an image.

## APPENDIX A – USE SCENARIOS WITH APPSWATCH

You can use your test scenarios with NRG Global's Appswatch for application performance monitoring. With Appswatch you can keep a close eye on an application's performance and ensure end users are getting acceptable response times. Appswatch gives you the tools needed to know if applications are running fast enough to keep customers from turning away or to keep employees and business processes working at peak productivity.

With Appswatch, you can:

- Receive alerts whenever an application scenario fails or underperforms - Appswatch shows you the actual screenshot of the failed screen!
- Automatically generate SLA reports, such as 10 slowest tasks per month, 10 least available tasks per month, etc.
- Perform trending analysis to see how an application's performance has improved or degraded over time

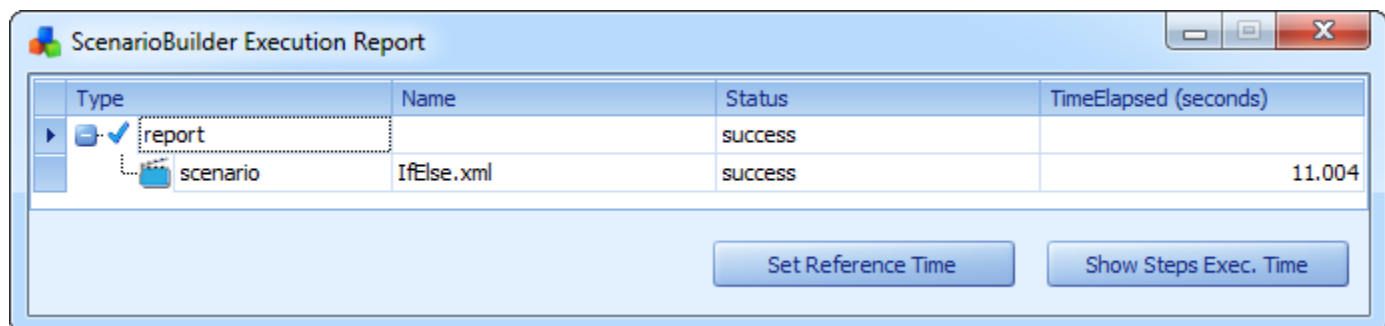
### APPSWATCH ARCHITECTURE

Appswatch accurately monitors response time from the user standpoint because robots on ScenarioStations repeatedly generate scenarios exactly as a real user would. ScenarioStations are situated at your key locations and report response times to the Appswatch Base. Organized views of data for troubleshooting, SLA reports, graphs, and alerting controls are available from the Appswatch Base.

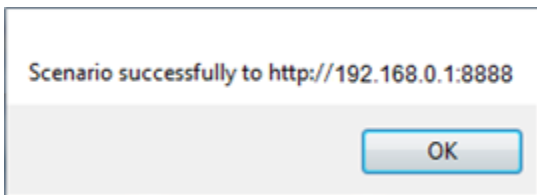
After completing and saving your test Scenario, send it to Appswatch Base. From there, send it to different ScenarioStations for execution. This distributed architecture enables you to send the same scenario to all your strategic locations, in order to test application performance from "end users" in New York, Chicago, San Francisco, or anywhere else. With Appswatch Base, you're able to manage all your tests from one centralized location.

### SEND A SCENARIO TO APPSWATCH BASE

Once you are finished composing your Scenario, click "Save". Test it using ScenarioBuilder's playback feature. Once it is finished running, the following window will pop up:



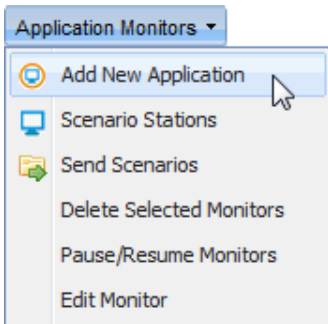
Click **Set Reference Time** and the scenario is ready to be sent to AppsWatch. Go to **File → Send Scenario to Controller** to send the Scenario to the AppsWatch Base. A window appears showing the status;



Once sent, scenarios can be added to Application Monitors in AppsWatch.

## ADD NEW APPLICATION MONITOR

From the “Application Monitors” menu, select “Add New Application”;



The “New Application” page opens in a new tab;

Application Monitors

+

Add New Application

Scenario Stations

Send Scenarios

Delete Selected Monitors

Pause/Resume Monitors

Edit Monitor

Required Parameters

Group: ?

Scenario Alias: ?

ScenarioStation: ?

Scenario: ?

Host/Server: ?

Events

+

Add Event

Type	Event	Profile	Reaction 1	Reaction 2
Warning	Scenario Resp. Time > 2min	24/7	emailme	

Description

Description: ?

Enter description...

Submit

Reset

**Group:** Select a Group from the drop-down list to establish the frequency – how often the scenario is executed (i.e. Every 5 minutes), and the profile – monitoring schedule (i.e. 24/7) associated with the Monitor.

**Scenario Alias:** Specify an Alias (name) for this Monitor

**ScenarioStation:** Select the ScenarioStation where this scenario will be played.

**Scenario:** Select the scenario which will be played.

**Host/Server:** Optional field used for advanced monitoring of System Outages

**Events:** Events trigger Alerts and Reactions when their conditions are met. To add an Event to your Monitor, complete the following parameters.

**Type:** Choose one of the following Alert types to be associated with this Event (ranked from least to most severe): Informational; Warning; Error; Down

**Event:** Select an Event from the drop down box. If the desired Event is not listed, you will have to **Add New Event**.

**Profile:** Select a Monitoring Window Profile from the drop down list. This determines the window during which Events will register (i.e. *Business Hours*). If the desired Profile is not listed, you will have to **Add Monitoring Window Profile**.

**Reaction 1 & Reaction 2:** Select up to two Reactions to be triggered by this Event (i.e. *Email me*). If the desired Reaction is not listed, you will have to **Add New Reaction**.

**Description:** Optional description field to further identify this Application Monitor

Complete the “New Monitor” parameters and “Submit”.

For a complete guide to AppsWatch, please download the AppsWatch User Guide at [www.nrgglobal.com](http://www.nrgglobal.com).

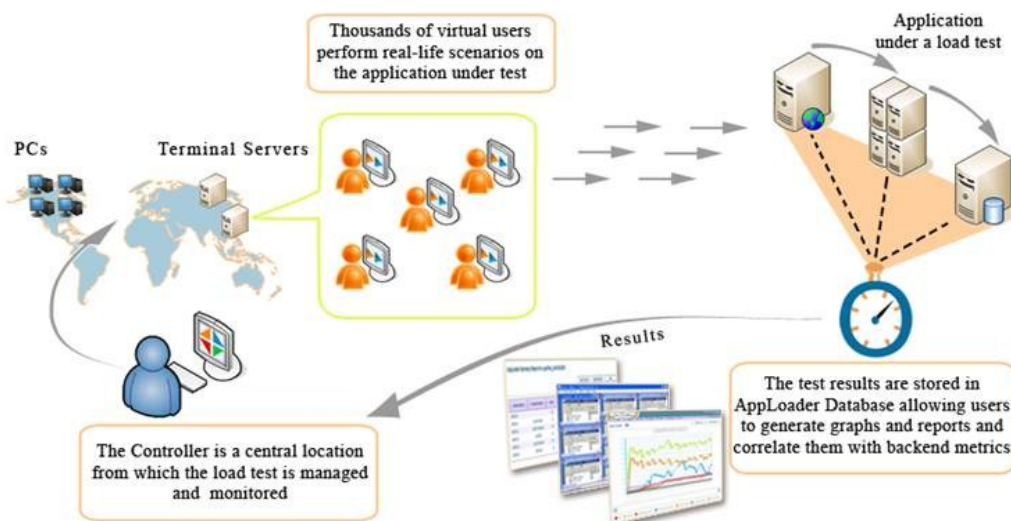
## APPENDIX B – USE SCENARIOS WITH APPLOADER

In conjunction with NRG Global's AppLoader, you can use your test scenarios to verify the performance of your application under varying load levels. AppLoader simulates real user load activity for applications, so you can be confident that a new or upgraded application meets expected performance requirements before bringing it into a live environment. With AppLoader, you can stress test the performance of your IT systems by simulating peak traffic from multiple locations simultaneously.

AppLoader gives you the answers to the following questions:

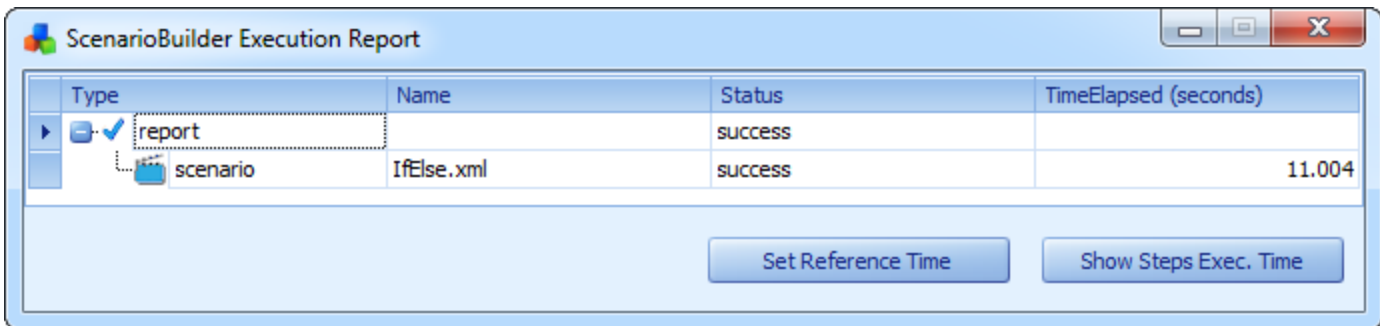
- Does your application perform properly under load?
- Do application response times meet your target requirements?
- How many simultaneous users can your application handle?
- AppLoader Architecture

The AppLoader Controller centralizes the administration of your load tests, and manages rUsers, terminal servers (Injectors), and scenarios. After you have created, tested, and saved your test scenario with ScenarioBuilder, you can send it to the AppLoader Controller. The Controller distributes the tests to the rUsers located on the terminal servers.

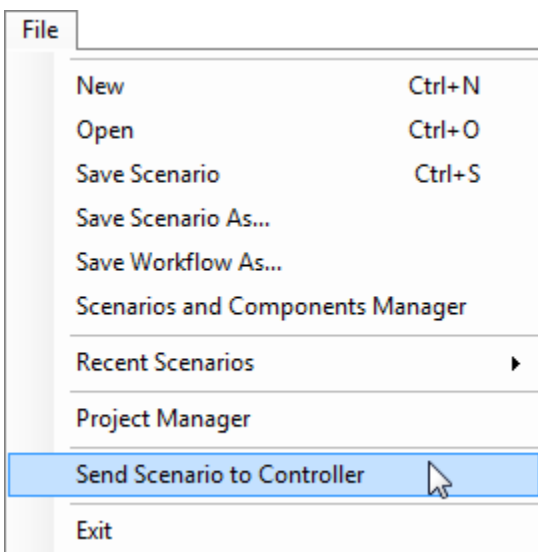


## SEND A SCENARIO TO APPLOADER CONTROLLER

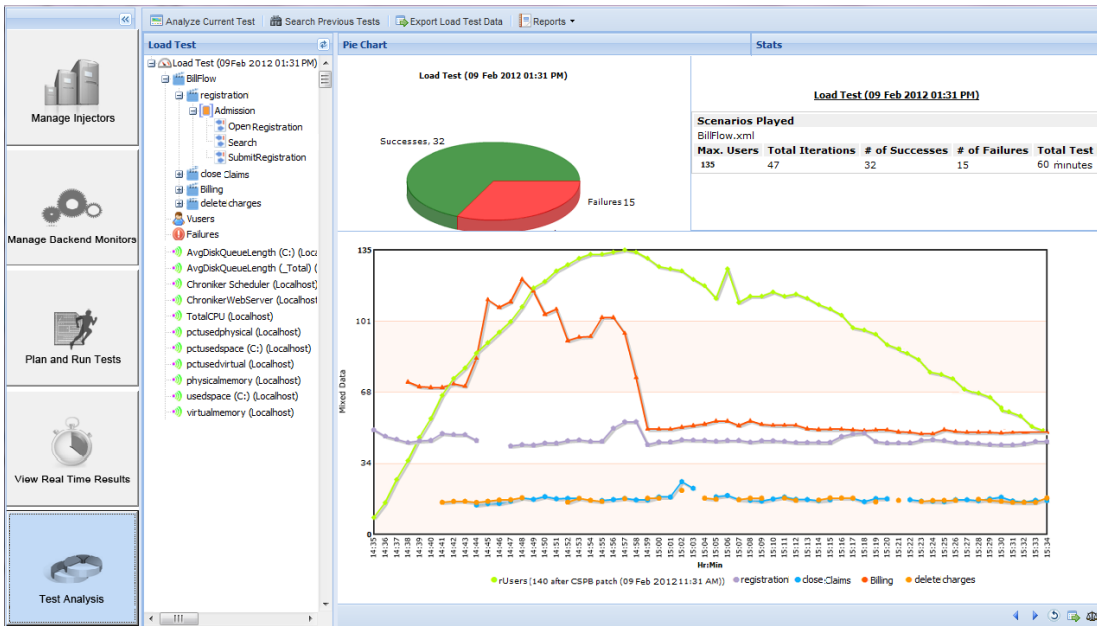
Once you are finished composing your Scenario, click **Save**. Test it using ScenarioBuilder's playback feature. Once it is finished running, the following window will pop up:



Click on **Set Reference Time** and the scenario is ready to be sent to Apploader. Go to **File→Send Scenario to Controller** to send the scenario to the Apploader Controller.



Once successfully sent, you can now manage these scenarios in AppLoader. Launch the web controller for AppLoader: <http://X.X.X.X:YYYY/> (where X.X.X.X is the IP address or hostname of the server AppLoader Controller is installed on and YYYY is the AppLoader port). The Controller guides you to create, run, monitor and analyze your test easily by following the menu on the left pane from top to bottom. You can see the easy to use structure of the Apploader below:



## MANAGE INJECTORS

The Load Injector is the Windows environment where rUsers are created and scenarios are played. Once the injector is installed, it will register itself with the AppLoader Controller.

From the “Mange Injectors” page, you may do the following:

Create rUsers

Start rUsers

Send Scenarios

Stop rUsers

Disconnect rUsers

Add New Injector

Injector Utils

Actions	Name	Host	Users Count	Location	Description
<div> <div></div> <div></div> <div></div> <div></div> </div>	NRG	192.168.1.1	200	NRG1	Injector added via VStation Installer

Click on **Create rUsers** to create local users on the Injector if Incremental mode used. Those are the real users that will be used for your load test. For domain users, select CSV mode.

Click on **Start rUsers** to generate the rUsers.

Now click on **Send Scenarios** to send the scenarios to different injectors.

Choose the scenario you wish to send from the drop-down menu and click send. The scenarios shown in the drop down are the ones sent from the ScenarioBuilder.

The 'Send Scenario to Injectors' dialog box is shown. It has a title bar with a close button. Inside, there is a label 'Scenario: ?' above a dropdown menu. The dropdown menu currently shows 'PatientRegistration.xml'. At the bottom right of the dialog is a 'Submit' button.

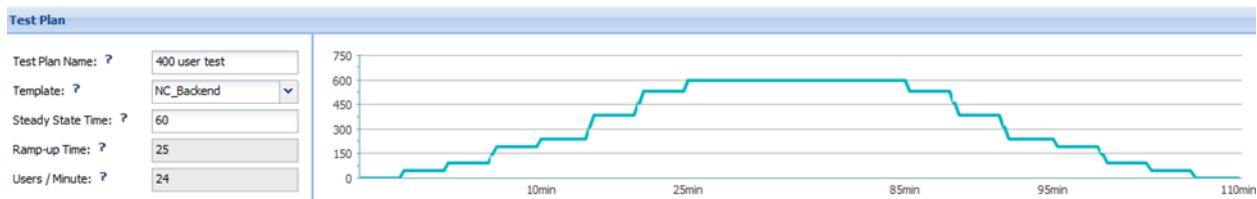
## RUN THE LOAD TEST

On the AppLoader Controller “Plan and Run Tests” page, click on [Create New Plan](#) to create a new Test Plan. A new form opens with the following sections:

**Details:** specify which Injectors to use, which Scenario(s) on each Injector, how many users on each Injector for the selected Scenario, delay (ramp-up) of rUsers on each Injector for the specified scenario.

Details				
Injector	Scenario	Users	Delay	Halt
.WEB1	registration.xml	300	5	<input checked="" type="checkbox"/>
.WEB2	Reports.xml	300	5	<input checked="" type="checkbox"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>

**Test Plan:** specify a meaningful test plan name, select a template if you have created one in the previous section and wish to use it, Input steady state time in minutes. Ramp-up time is automatically calculated after entering the information above. Ramp-up time is the maximum time taken from the first rUser injected to the system to the time the last rUser starts playing. Users/Min is also automatically calculated. This is the maximum number of users/min injected during the ramp-up time.



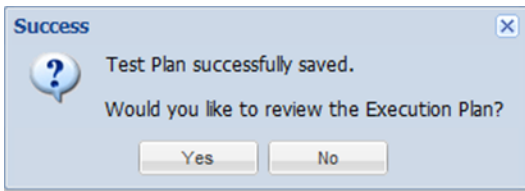
**Scenarios:** Each scenario selected in the Details section above will show in the list below along with its duration in minutes. The number of iterations are calculated from the steady state time and ramp-up time specified in the Test Plan section above and the duration of the scenario.

Scenarios		
Scenario	Duration (Min.)	Iterations
registration.xml	30	3
Reports.xml	52	2

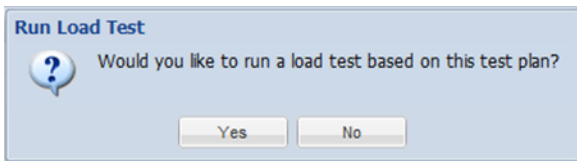
**CSV files:** All CSV files assigned to the Scenarios selected in Details section will show below. This section allows you to parameterize your test data entry. You can specify the Access Rule, Start Line and how many rows you would like assigned for each user by double clicking on the field and entering the desired values.

CSV Files				
Scenario	CSV File	Access Rule	Start Line	Rows / User
registration.xml	variablesA.csv	Sequential	1	1
Reports.xml	Variables.csv	<div> <input type="text" value="Sequential"/> </div> <div> <input type="text" value="Unique"/> <input type="text" value="Sequential"/> <input type="text" value="Random"/> </div>	1	1

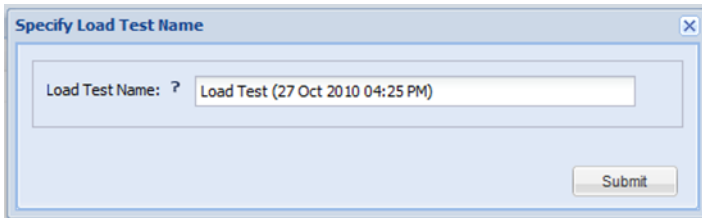
Once you filled all the needed sections, click on  **Save** button. The below message will show:



Click **Yes** to verify the execution report based on the information you supplied in the Test Plan. The execution report will show you each rUser name, host, assigned Scenario, assigned iterations and start delay. Then once you minimize or close the execution report, another message will show as below.



Click **Yes** to run your test or click **No** if you like to run it later from the button at the top of the Plan and Run your Test Page. If you clicked **Yes**, another popup will show to name your load test. Once you have named your test, click **Submit**.



Your load test is now running and you can go to the View Real Time Results Page to view and interact with the test in real-time.

For a complete guide to AppLoader, please download the AppLoader User Guide at [www.nrgglobal.com](http://www.nrgglobal.com).

## INDEX

### Actions

- Action, 17
- Actions Pane, 22
- Context Menu, 68
- Add New Application Monitor, *(in AppsWatch)*, 125
- All Windows Action, **111**
- Anchor Position, 105
- AppLoader, 127
- AppsWatch, 124
- Begin Transaction, **101**
- Best Practice
  - Find Image "Below the Fold", 52
  - Keep Scenario Synchronized with Application, 50
  - Scenario Design Basics, 33
- Black and White Contrast Ratio, 56, 105
- Blank, *(begin Scenario with)*, 21
- CallDll, **117**
- Capture Image, **118**
- Case, **113**
- Case/Switch, 72
- Child Action, 53, 69
- Click on Application Icon, *(begin Scenario with)*, 20
- Click on Image, **104**
- Clipboard, 23, 98
- Close Application, **100**
- Comment, **117**
- Components, 41
  - Component, 17
  - Components Pane, 23
  - Copy, 44
  - Define, 42
  - Export, 45
  - Import, 46
  - Save, 43
- Define Component, **102**
- Description Pane, 23
- End Transaction, **101**
- End User License Agreement, 15
- Execute Component, **101**
- Execution Status Variable, 62, 71, 105
- Exit, **115**
- Favorites, 97
- Find Image, **105**
- Find Linked Images, **106**
- Function Keys, **107**
- Hotkeys, 122
- If, 63, 69, 71, **112**
- Image Actions, 50
- Image Variations, 56
- Image Preferences, *(for Scenario Recorder)*, 122
- Image Search Area, 58, 105
- Increment a Variable, 61, **113**
- Install ScenarioBuilder Software, 7
- Label, **115**
- Line Filler, **117**
- Log Level, 86
- Loop, 52, 62, **113**
- Mouse Click, **107**
- Mouse Move, **108**
- Move Window, **109**
- On Failure, 17, 75, **116**
- Open Web App, *(begin Scenario with)*, 20
- Open with File Association, **103**
- Play Scenario, **104**
- Preferences, 121
- Project's Scripts
  - Project's Scripts Pane, 23
- Projects, 82
  - Change Active, 84
  - Create, 89
  - Project, 17
- Properties Pane, 23
- Raise Failure, **116**
- Read Serial Port, **117**
- Read Text into a Variable, **114**
- Rendezvous Point, **115**
- Resize Window, **110**
- Response Time, 17
- Resume, **116**
- Run Command Line, **103**
- Scenario, 17
  - Export, 91
  - Import, 92
  - Nesting, 94
  - New Scenario, 19
  - Properties, 81
  - Workflows, 96
- Scenario Parameters, 60
  - Attach a CSV File, 66
  - Call Variable, 67
  - Create CSV File, 64
  - Edit CSV File, 65
  - Last Line Used in CSV File, 67
- Scenario Recorder, 26
- Scenario Samples
  - Create First Scenario with Scenario Recorder, 29

Create First Scenario with ScenarioBuilder, 37  
Create Gmail Account (*Advanced*), 76  
in "Help" Menu, 54  
Scenario Window, 22  
Send a Scenario to AppLoader Controller, 127  
Send a Scenario to AppsWatch Base, 124  
Set a Variable, 60, **114**  
Skip To, **116**  
Sleep Time, 105  
Splash Screen, 18  
Start Application, **100**  
Start Application, (*begin Scenario with*), 20  
Support, 7  
Switch, **113**  
Theme, 85  
Timer, **116**  
Tool Bar, 24  
Transaction, 17  
Type Text, **106**  
User Interface, 22  
    Default Layout, 22  
View, 23  
Wait, **117**  
Wait for Image to Disappear, 54, 105  
Window Actions, **109**  
Write Serial Port, **118**