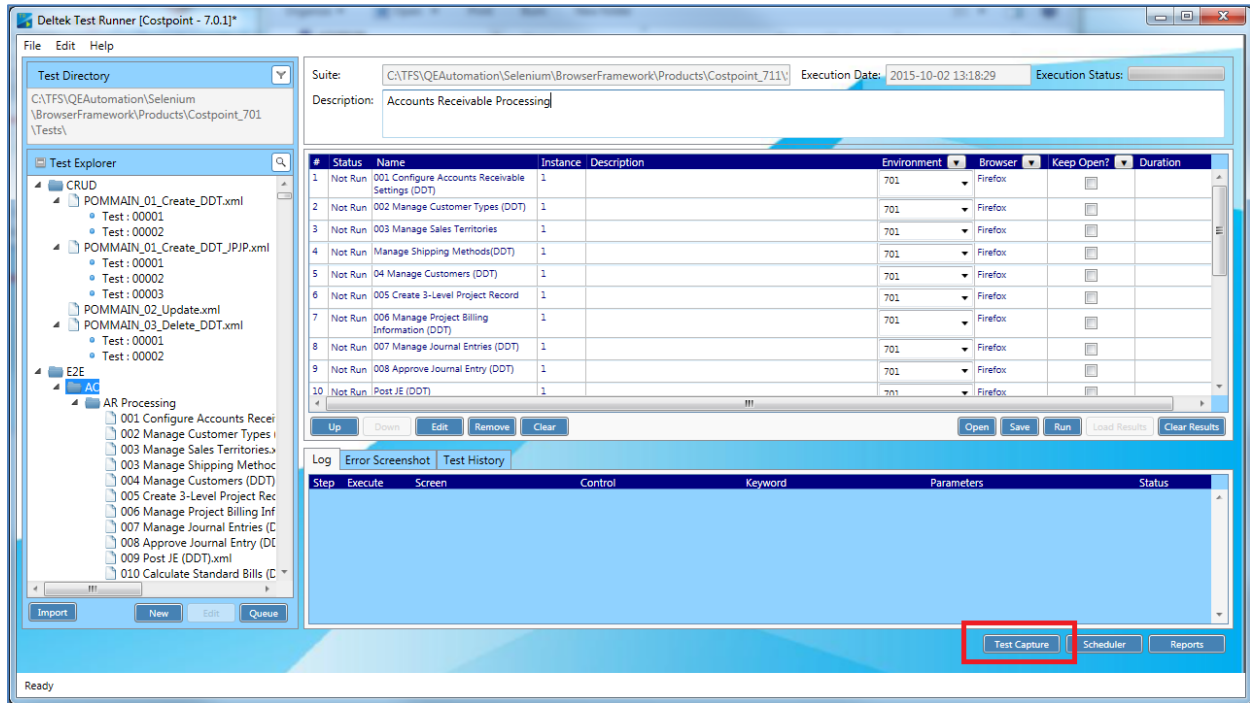
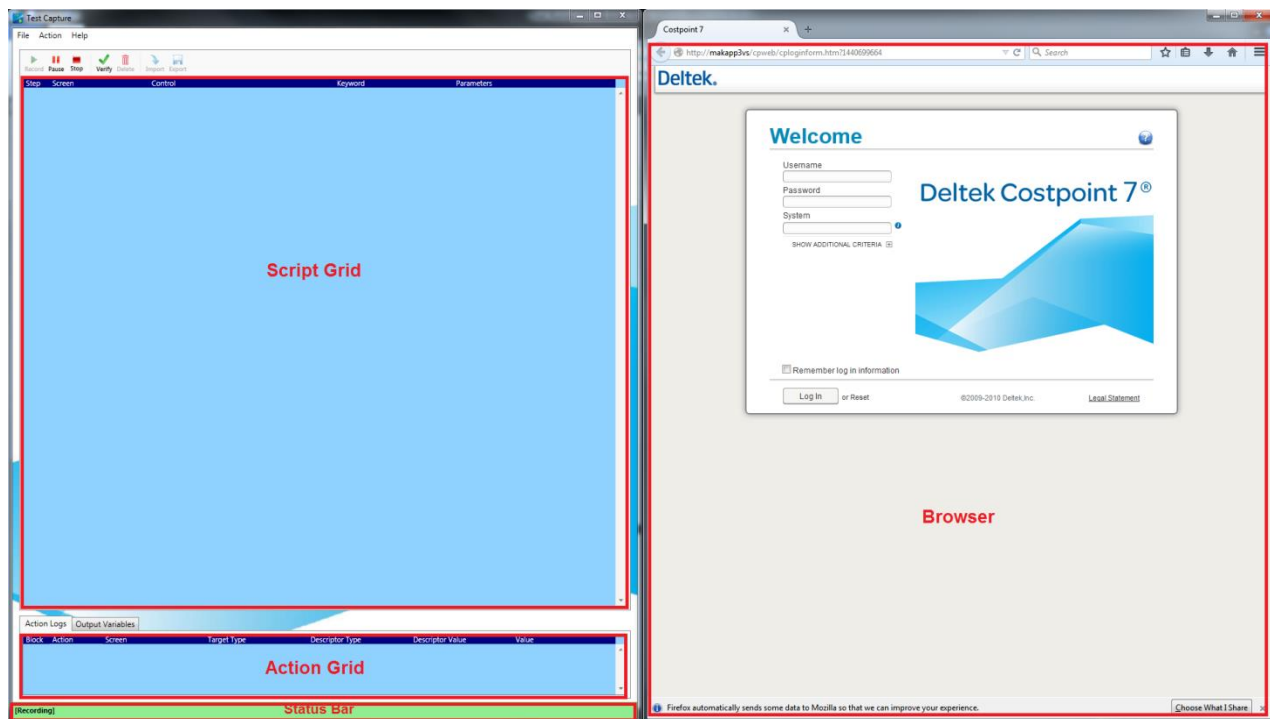


Test Capture Quick Reference Guide

The *Test Capture* feature allows the user to record simple actions or verify some elements from *Costpoint 7* web application and convert them into a test script. Click the *Test Capture* button found at bottom right corner of the Test Runner window to access *Test Capture*:



Test Capture



Test Capture (left) and the browser (right)

Below are the simple usage instructions of this feature. It is also advised to take a look at *Appendix C: Test Capture Limitations* for additional information about the feature.

Script Grid

This grid displays the test script created during the recording.

Action Grid

This grid contains the actions being recorded for tracing. Captured clicks and keystrokes are shown here.

Commands

Record

This will open a new instance of Mozilla Firefox and start the recording of the test. After clicking this button, clicks and keystrokes performed on the web application will be captured and converted into test steps. The Script grid and the Action grid will be populated as long as the recording is ongoing.

Start Browser

This will start a new instance of Mozilla Firefox. This is used for instances where the user wants to record a specific set of steps without having to go through the login steps. It can be accessed through the menu bar: *Actions > Start Browser*.

Verify

This will verify the current value of the selected element.

Delete

This will delete current selected step. The user can only perform step deletion if Test Capture is in Paused state.

Pause

The recording will be paused. Actions on the browser will not be recorded until the *Record* button is pressed.

Stop

The recording will be stopped and the browser will be closed. Clicking the *Record* button after stopping the recording will start a new recording and clear the current test script.

Clear

This will clear the entire recorded test script.

Import

This will import an existing test script into test capture. The user will be given option to playback imported steps when resuming test recording.

Export

This will save the recorded test script into XML format. The exported test script may be further edited in the *Test Editor*.

Status Bar

The status bar provides the user information on the ongoing status of the recording. The colors of the status bar have significant meanings that the user must be aware of:

[Recording]

Green – The test recording is ready to accept any user action.

[Ready to Verify]

Yellow – The test recording is in *Verify* state. Clicking on an element will verify its current value.

[Wait]

Light Yellow – Signal for the user to wait before performing another action.

[Paused]

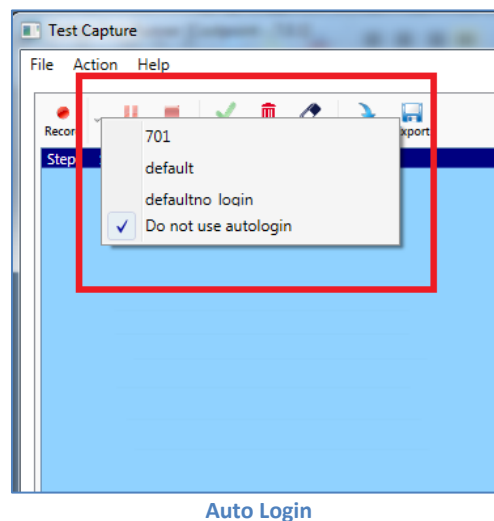
Orange – The test recording is paused.

[Info]

Pink – Displays information messages (or errors) to the user

Auto Login

The user can opt to perform automatic login to Costpoint, by selecting any of the *Environments* defined in the *Settings* form.



Recording will only commence, as soon as the automatic login finishes. The default setting is 'Do not use autologin'.

Alternate Keywords: ALT+Click

Alternate keywords are control keywords accessible to users in addition to 1 action keyword (ex: *TextBox* -> *Set*) and 1 verify keyword (ex: *TextBox* -> *VerifyText*) that are available by default.

For now, only 2 control types make use of alternate keywords: *TextBox* and *Table* controls.

Control Type	Action Keyword	Verify Keyword	Alternate Keyword
TextBox	Set	VerifyText	<i>ClickTextBoxButton</i>
Table	SetTableCellValue	VerifyTableCellValue	<i>GetTableRowWithColumnValue</i>

To access alternate keywords, instead of the just clicking on a page element, use **ALT+Click**.

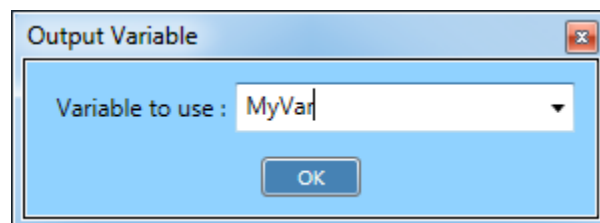
Output Variables

Some control keywords make use of *Output Variables* as parameters. These variables are used to store values retrieved by these keyword and can be accessed by succeeding steps via the **O{OutputVariable}** notation.

One of the keywords that make use of an output variable is *GetTableRowWithColumnValue* keyword of the *Table* control:

KEYWORD	Description	Parameters
GetTableRowWithColumnValue	Assigns row number that matches Value under the column ColumnHeader to VariableName	ColumnHeader Value VariableName

When the alternate keyword for the *Table* control (*GetTableRowWithColumnValue*) is accessed by the user using ALT+Click, a dialog will be displayed to get from user the output variable name to be used for the keyword:



Output Variable dialog

Once a name has been provided by the user, the output variable (*MyVar*) will be used as the *VariableName* parameter in the *GetTableRowWithColumnValue* keyword (Step 3).

All output variables created by user via the *GetTableRowWithColumnValue* keyword will be listed in the *Output Variables* tab.

The image shows two side-by-side windows. The left window is 'Test Capture' and the right is 'Costpoint 7.1.1 - Company 1/O...'. Red lines connect elements between the two windows.

Test Capture Window:

Step	Screen	Control	Keyword	Parameters
1	CPMain	NewMenu	SelectMenu	Materials-Product Definition-Items-Manage Parts
2	POMBART	MyForm	ClickButton	Title
3	POMBART	MyFormTable	GetTableWithColumnValue	Part Security*** UNRESTRICTED - MyVar
4	POMBART	MyFormTable	VerifyTableCellValue	COMyVar - #16 GA MTW GRN

At the bottom of the Test Capture window, the 'Output Variables' tab is active, showing:

Active	Name	Current Value
<input checked="" type="checkbox"/>	MyVar	6

Costpoint 7.1.1 Window:

The 'Manage Parts' table is displayed with the following data:

Part ID	Rev	Description	Part Security	U/M	Status
3108C28-20		BOLT-4 REF NAS3108C28-20	Unrestricted	EA	Release
3108C28-24		U-BOLT REF NAS3108C28-24	Unrestricted	EA	Release
62828-90038-1		AC LINE FILTER	Unrestricted	EA	Release
#16 GA MTW BLU		16 GA BLUE WIRES	Unrestricted	FT	Release
#16 GA MTW GRN		16 GA GREEN WIRE	Unrestricted	EA	Release
#16 GA MTW GRN1		16 GA GREEN WIRE	Unrestricted	EA	Release
#16 GA MTW RED		16 GA RED WIRE	Unrestricted	FT	Release
#16 GA MTW WHT		16 GA WHITE WIRE	Unrestricted	FT	Release
#2 THHN BLACK		ELECTRICAL WIRE	Unrestricted	EA	Release
#20		8 1/2" X 11" WHITE COPY PAPER	Unrestricted	CTN	Pre-Rel
#20		#20 ITEM	Unrestricted	EA	Release
#24246	1	#24246#24246	Unrestricted	EA	Release
#3		CLAMPGROUND	Unrestricted	EA	Release
#3		part for #3 again	Unrestricted	EA	Release
#8 THHN GREEN		ELECTRICAL WIRE	Unrestricted	EA	Release
		apostrophe	Unrestricted	EA	Release

Red lines indicate the flow of data: from the 'Part Security' column in the table to the 'Parameters' field of Step 3, and from the 'Current Value' of MyVar to the 'Parameters' field of Step 4.

Using OutputVariables with GetTableWithColumnValue keyword

Notice that MyVar was also used for any succeeding step working with the same Table control, like in Step 4 where the VerifyTableCellValue keyword was used, because MyVar's Active checkbox is checked in the Output Variables tab. If user wishes to use the actual value (6), instead of the MyVar, he simply unchecks the Active checkbox.

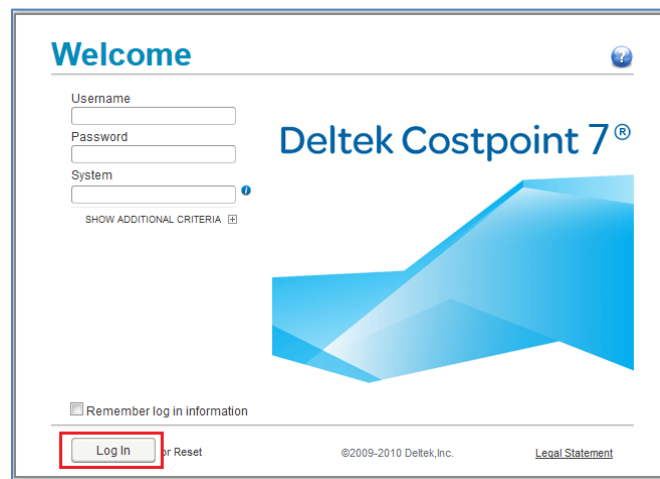
For now, only the Table control type makes use of output variables.

Test Capture Limitations

1) 'Clickable' controls that disappear from screen on click

The test capture feature works with the use of injection of *javascript (JS)* codes to the *Document Object Model (DOM)*. These *JS* codes are designed to identify web elements within the *DOM* that the user clicked or typed into. With buttons and other controls that 'disappear' upon user click, the *JS* code sometimes fail to retrieve the web element before it is removed from the *DOM* due to the web page transitioning to another state because of the user click.

An example is the *Login* button:



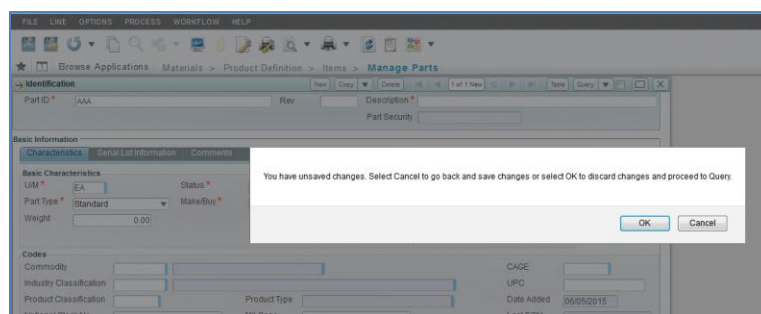
Login button

When the user clicks on the *Login* button, the page transitions from the *Login* page to the *Main* page of *Costpoint*, refreshing the browser *DOM*. This refreshed *DOM* no longer contains the *Login* button web element, thus the recorder fails to record the corresponding user action performed on the login button.

As an alternative, the user can press the **left mouse button and not release (mouse-down)** until the *Click* step is recorded in the *Script Grid*, and then **move the mouse cursor away from the control** to prevent firing of the *Click* and any attached actions programmed into the control when clicked. The user can then re-click the control to proceed with recording.

2) Unexpected dialogs

Since the test capture feature implements an engine that continuously monitors web page activity, it is essential that the browser *DOM* is always accessible. Unexpected modal or blocking dialogs prevent the tool from accessing the *DOM* and record user actions. Below is an example:

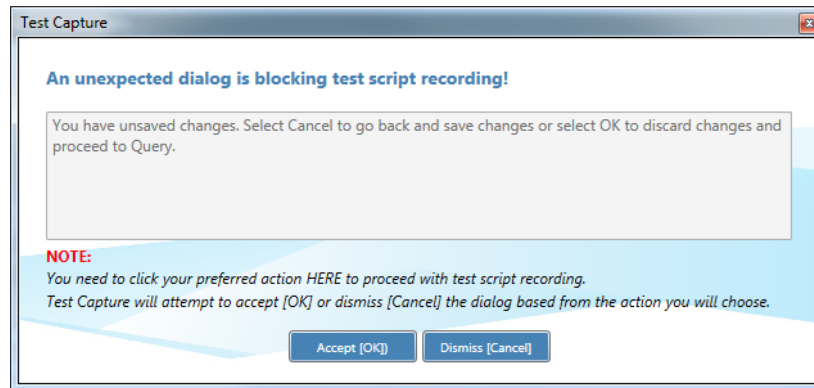


Unexpected Dialog

Some examples of controls/actions in *Costpoint* that trigger this include but are not limited to:

- Clicking on *File->Close Application* when there is a pending change to current record
- Clicking on *Query* when there is a pending change to current record

Once an unexpected dialog (*Unexpected Dialog*) is encountered, the tool will display a dialog message of its own (*Test Capture Dialog*), which mirrors the message of the unexpected dialog (*Unexpected Dialog*):

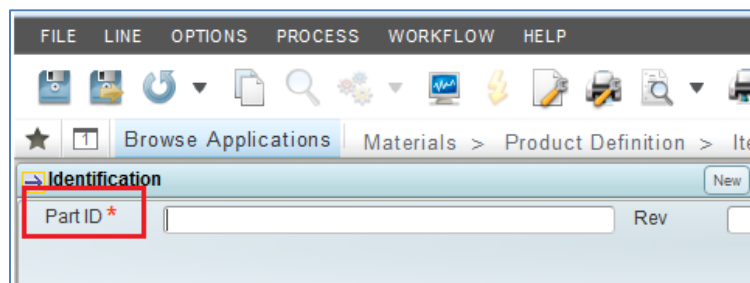


Test Capture Dialog

The user should use *Test Capture Dialog* instead of *Unexpected Dialog* to click on the desired response for the recording to proceed. Failure to follow this routine will end the test recording.

3) Label controls not supported.. yet

Currently, actions performed against *Label* controls are not yet supported for recording.



Label Control