SOASTA

TouchTest™ Web Tutorial

SOASTA CloudTest® TouchTest™ Web Tutorial

©2012, SOASTA, Inc. All rights reserved.

The names of actual companies and products mentioned herein may be the trademarks of their respective companies.

This document is for informational purposes only. SOASTA makes no warranties, express or implied, as to the information contained within this document

Table of Contents

Why Mobile Web Testing? 1

CloudTest® Basics 1

What Does Touch Test™ Web Record? 2

TouchTest Web Prerequisites 2

Registering Your Device to Use TouchTest™ Web 3

Approving a Mobile Device (Device Administrators/CloudTest Lite Users) 7

Create a TouchTest™ Web Clip 8

Create a Composition 11

Result Details 13

Advanced Clip Editing 16

Inspecting Browser Action Details 17

Browser Action Properties 19

Adding Outputs 20

Examining Outputs in Results 21

Adding an Element Validation 23

Adding a Text Validation 25

Adding an Interval Delay between Each Action 26

Analyzing Validations in Results 27

SOASTA

Why Mobile Web Testing?

CloudTest® Mobile, featuring TouchTest™ technology, delivers for the first time, complete functional test automation for mobile web sites using browser on your iOS device. TouchTest™ technology delivers fast, precision functional testing while increasing the stability of automated tests across releases.

The TouchTest™ Web app makes web sites viewed with iOS browser testable. TouchTest Web is launched using a lightweight web service called TouchTest™ Agent. Devices can be dedicated to testing in the lab, used as part of an external test, or crowd-sourced as part of a high volume, globally distributed test.

CloudTest Mobile support is provided for recording user actions within any iOS 5.0 or greater device including iPhone, iPad, and iPod Touch. There is no need to jailbreak the iOS device and the device can be untethered.

CloudTest® Basics

SOASTA CloudTest® provides fast, effective performance, load and functional test automation of any modern Web application, Web service, or mobile application in a lab, staging or production environment using unique visual programming and multi-track user interfaces. The CloudTest platform can utilize both public and private cloud resources to assure any web or mobile application won't fail under peak user traffic.

Central is the primary user interface in which CloudTest® runs, and contains CloudTest's primary features, organized by sections.

Central's first section—highlighted in light blue—contains the Welcome page, and the primary test building tools.

The **Composition** is the test itself as presented in the **Composition Editor**, and contains one or more **Clips** arranged on **Tracks** and governed by user-specified sequence and tempo. The Composition Editor is a player, debugger, and the dashboard where results are analyzed.

The **Clip** is the basic building block of a test as built in the **Clip Editor** and has a **Target** such as a web site, or in the case of TouchTest™, a mobile app. A clip can be thought of as a visual script that is composed of a series of timed or sequenced

events, which correspond to gestures performed on the mobile device. It can contain messages, app or browser actions, and scripts, as well as delays and checkpoints—all of which can be organized into containers (i.e. groups, chains, transactions, etc.)—and parameterized as required.

CloudTest Mobile clips are recorded directly from the mobile app and added to the Clip as you perform them on the mobile device.

What Does Touch Test™ Web Record?

TouchTest™ Web records the details of actual browser gestures and events on the mobile web site under test. These gestures and events are represented within the CloudTest/Lite **Clip Editor** as Browser Actions.

Each gesture you perform on a TouchTest Web-enabled device is precisely, and automatically, added to the test clip as a browser action.

Like any clip element within CloudTest®, browser actions have attributes such as inputs and outputs, properties, waits, and validations that can be parameterized. Additionally, a browser action can be added to any containers (e.g. if-thenelse, switch, chaings, transactions, and groups).

TouchTest Web Prerequisites

Touch Test™ Web has the following prerequisites:

1. An iOS device running iOS 5.0 or later.
2. A CloudTest user account on any CloudTest Edition, including a CloudTest Lite instance (with its default user account).

   TIP: CloudTest Lite is a free downloadable edition of SOASTA CloudTest available as a VMWare® image that can be run in any virtual machine software that can open it. CloudTest Lite provides one free user account that can be used with one mobile device. The TouchTest Web app must be present on the mobile device.
3. The TouchTest Web app installed on the mobile device that will be used to test. If you haven't already done so, open the App Store on your iOS device and download TouchTest™ Web.
4. Once the Touch Test™ Web app is installed and launched, tap Go to open the TouchTest Agent registration page.

Registering Your Device to Use TouchTest™ Web

The TouchTest™ Agent is responsible for launching the apps that are being tested on a given device. It is a web application that is runs in mobile browser on iOS devices. To get started, browse to the TouchTest Agent URL below on the mobile device and then perform the one-time registration steps that will enable your device for use with TouchTest.

Note: If you clear your cookies on the given mobile device after registration, you may need to register your device again so that TouchTest can recognize it. This does not consume an additional license.

1. Click Go to launch the TouchTest Agent URL.

   Optionally, specify a different TouchTest Agent URL before pressing Go. The default TouchTest Agent URL is.

   http://demo.soasta.com/concerto/touchtest

The following screen appears.

2. Login with your SOASTA CloudTest user name and password.

   Note: For a Simulator, first click the link below the Login button.

If the device is not already registered, the Register Device page below appears.

3. Click the Register Device button to continue.
    a. First, the Install Profile screen appears. Click the Install button to proceed.

b. The Unsigned Profile alert appears to indicate that mobile device settings will be changed. Click Install Now to proceed.

c. If a passcode is in effect on the mobile device, an additional prompt will appear to authorize the profile installation.

4. When prompted, give the TouchTest Agent a name. For example *SOASTA Demo iPad*. Note that this name will be used throughout the product to refer to this device. You can rename your device agent at any time by locating it in the Central > Device Agents list and using the right-click Rename command).
5. Once this name is entered, click Submit for administrator approval.

Once the request for Administrator approval has been made, the TouchTest Agent will continue to poll CloudTest for approval. If you are a CloudTest Lite user, you are also the Mobile Device Admin and you should approve the device yourself using the steps in the following section.

Note: It is not necessary to keep the TouchTest Agent running while this approval is pending. The TouchTest Agent will resume polling for its approval once restarted.

Once your TouchTest agent is approved by the Mobile Device Administrator, the Connected page will appear the first time TouchTest is launched in browser on the approved device. On subsequent launches click Login to Connect and Logout to Disconnect.

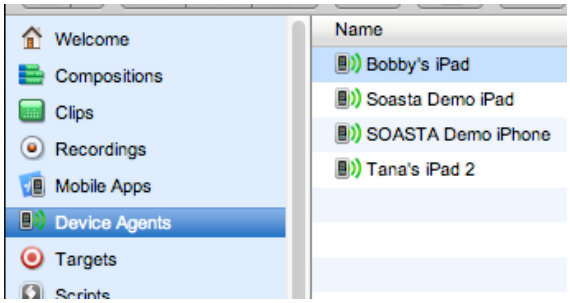
If your device is rejected by the Administrator, that status will display.

Approving a Mobile Device (Device Administrators/CloudTest Lite Users)

The TouchTest™ Mobile Device Administrator has the responsibility to approve or reject the devices attempting to join testing. Administrators will use the following steps to approve/reject the devices attempting to join.

> **Note:** In CloudTest Lite the default user is also the Mobile Device Administrator.

1. Login as the user with mobile device administrative rights.
2. Click Central > Device Agents



The Device Agents list displays all the devices in queue by name. Additionally, the model (iOS device type), OS (iOS version), and current status of the TouchTest agents are displayed in the list columns.

Those devices that have the status Pending Approval need administrative attention. For example, the SOASTA Demo iPhone shown in the list above.

3. Click Approve to complete adding a device and Reject to deny its access.

> **Note:** Once a device is approved it cannot be deleted without contacting SOASTA Support. CloudTest Lite users may approve their single device but will also need to contact SOASTA Support to delete.

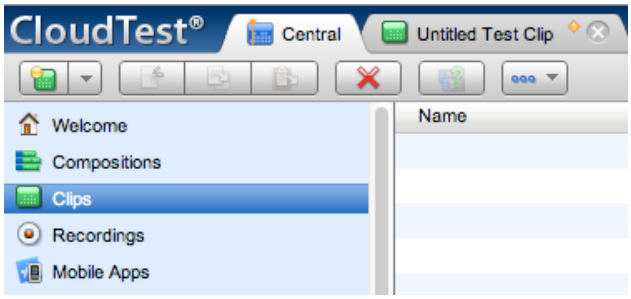
Create a TouchTest™ Web Clip

Now, we can create a new clip that will be used to perform mobile web recording and serve as the basis for your TouchTest.

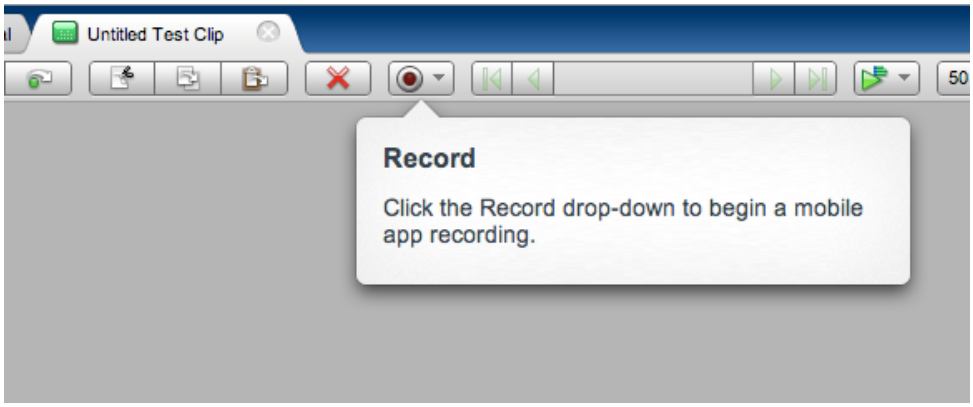
- Login to CloudTest on your desktop computer.
- Open the TouchTest Agent in browser on your mobile device, and click Login (your previously entered username and password will be auto-populated).

Once successfully logged on, the device Status will be *Connected*.

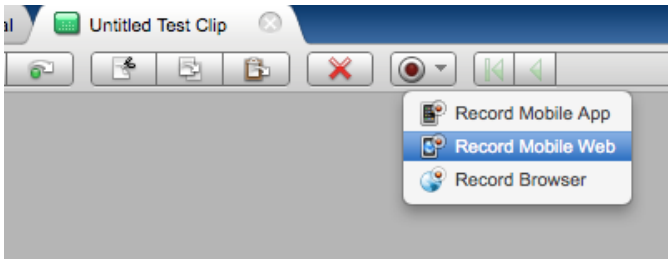
1. In the CloudTest, Central tab, select Clips, and then click New on the toolbar.



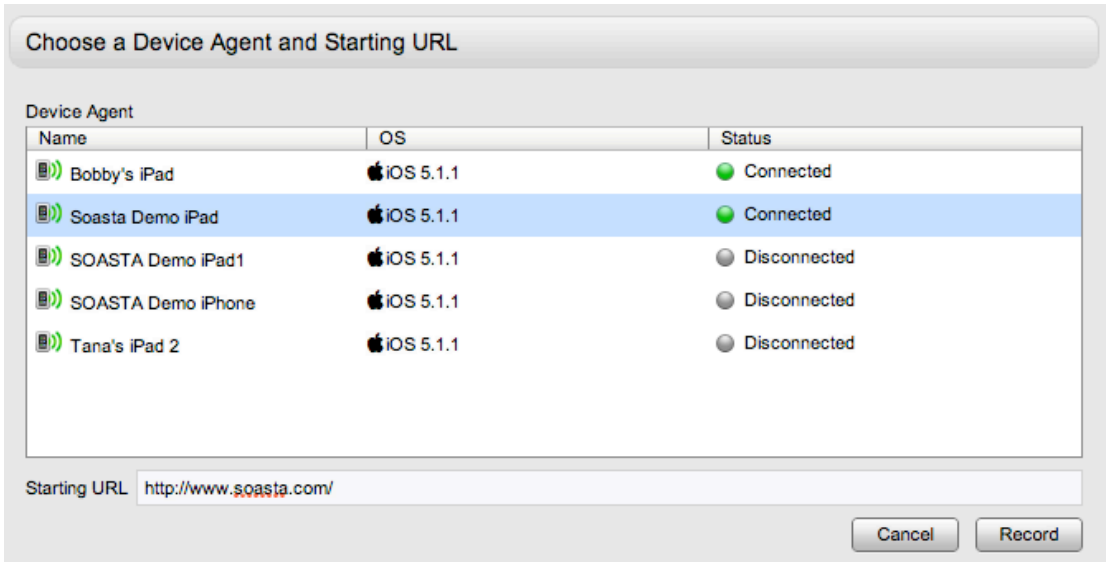
A new *Untitled Test Clip* opens in a Clip Editor tab. A Record pop-up identifies the Record drop-down.



2. Click the Record button and then select Record Mobile Web.



The Choose a Device Agent and Starting URL box appears.



3. Select the TouchTest Agent that you created above. For example, *SOASTA Demo iPad*.

Specify the Starting URL of the mobile web site to record.

For example, enter the URL for the SOASTA web site:

http://www.soasta.com/

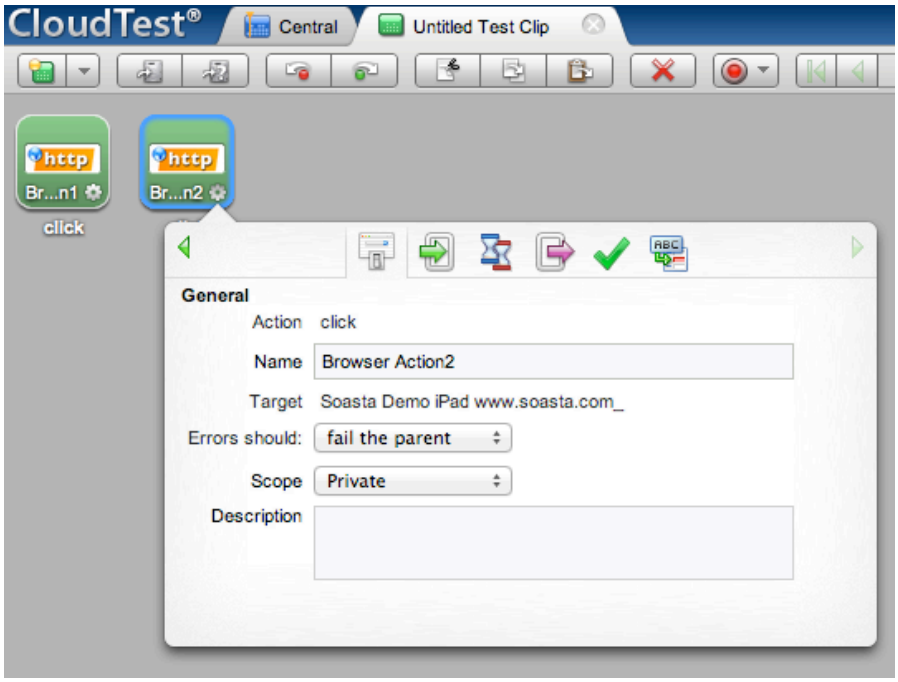
4. Once the Device Agent selected is “Connected” and the Starting URL has been entered, the Record button in the Choose box becomes active. Click it to continue.

   TouchTest Agent launches TouchTest Web, after which the target site appears in a browser window.

5. Perform the planned mobile app user interactions on your mobile device. For each action you perform in the browser, TouchTest Web adds an action to the clip.

   For example, in our demo clip, which is used in the remainder of this tutorial, we tapped the following sequence:

   a. On SOASTA home, in the What’s New section, click the first link.
   b. Click the SOASTA logo (top left).
   c. Under the Customers section, click the customer at the top.
   d. On the navigation menu, tap Products and then tap CloudTest Lite.
   e. On the navigation menu, tap Customers and wait for the page to load.
   f. On the navigation menu, tap Knowledge Center and wait for the page to load.



The Info Window switches along the stream of actions as they are added. This window provides a convenient entry point to action details and can be used to change properties and add accessors such as validations. Actions can also be double-clicked to open them in the Clip Editor lower panel.

TIP: You can hide the Info Window using the Etc. (…) menu Show/Hide Info While Recording command. Click the Gear icon to Show Info for editing, or alternately, double-click to use the Action Editor in the lower panel.

6. Once the relevant interactions have been recorded, click the Record button again to stop the recording.

Create a Composition

With your test clip open in the Clip Editor, you are ready to create and play this simple test clip in a new test composition.

1. To create a new composition from your test clip, click the Use in Test Composition drop-down in the upper-right corner of the Clip Editor toolbar.

   Before making a selection note the menu commands:

   - **Open in Test Composition**

     Choose Open in Test Composition to add this clip to a new draft composition where additional composition parameters can be set in the Composition Editor, Edit tab before proceeding to play.

     This option is useful when composition-level properties need to set, and other composition-building tasks done before the test is ready to play.

   - **Play in Test Composition**

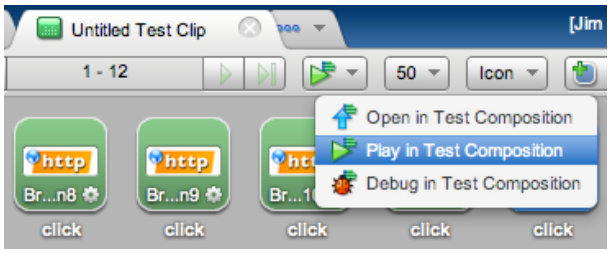
     Choose Play in Test Composition to add this clip to a new draft composition where it will immediately be played in the Composition, Play tab before proceeding to edit parameters or play.

     This option is useful for initial runs of a test prior to adding any necessary composition-level properties.

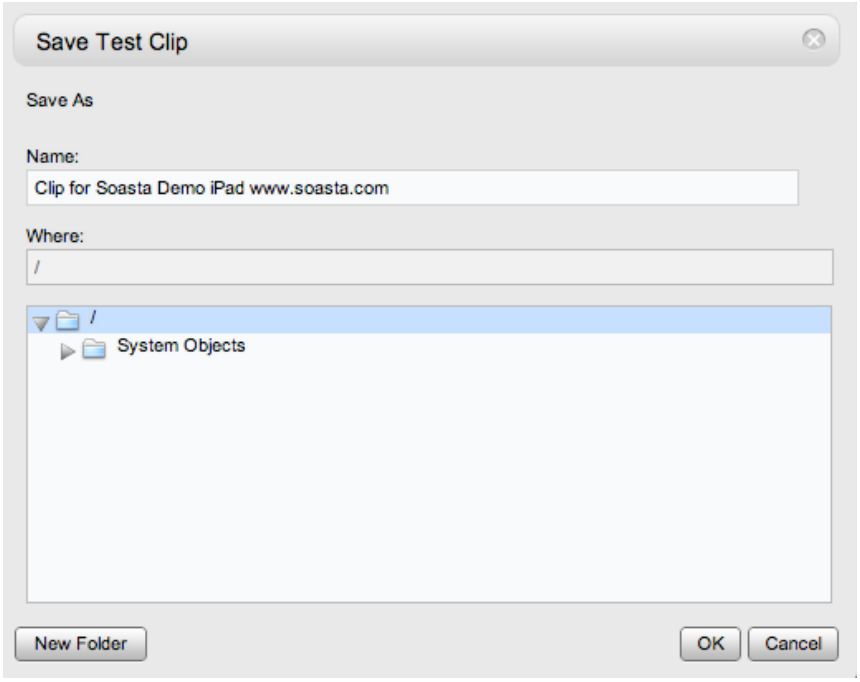
   - **Debug in Test Composition**

Choose Debug in Test Composition to add this clip to a new draft composition where it can be debugged in the Composition, Debugging tab before proceeding to edit parameters or play based on debug actions.

2. After observing the menu commands, choose Play in Test Composition.



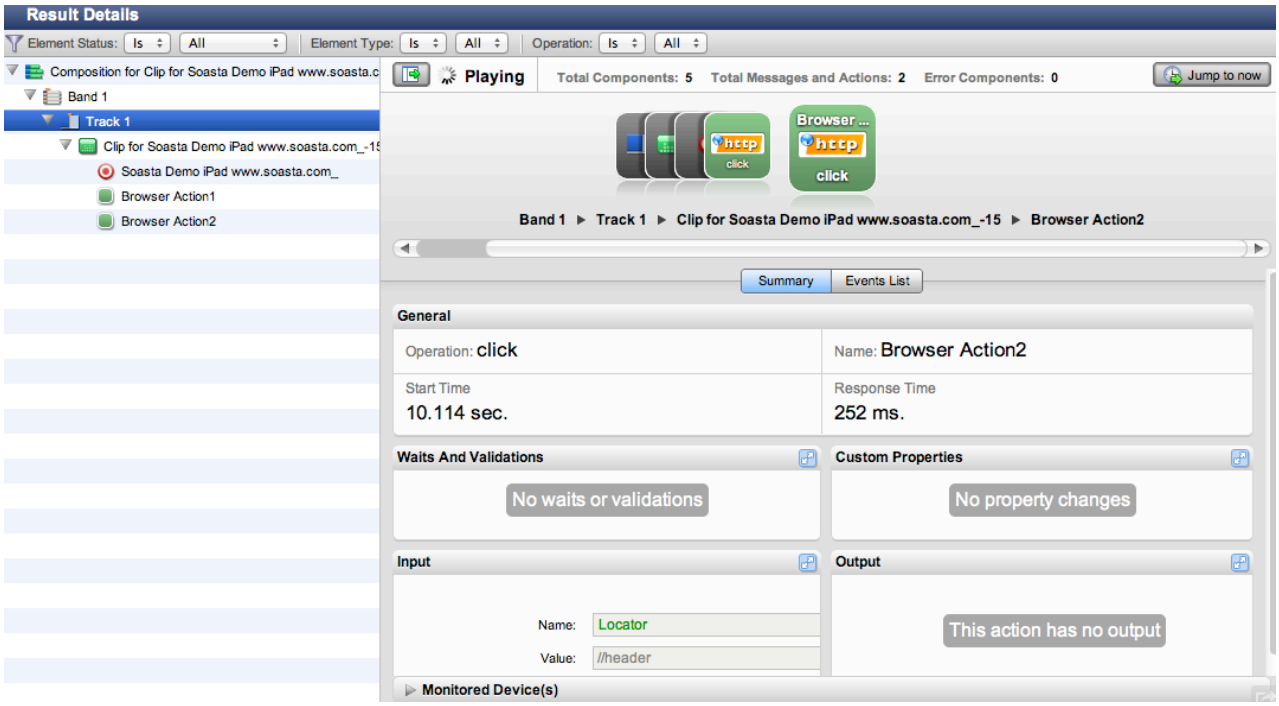
3. The Save Test Clip box appears. Accept the default test clip name and click OK.



4. A Composition Editor tab appears. As play starts, the Composition Editor switches immediately to the Play tab.

Result Details

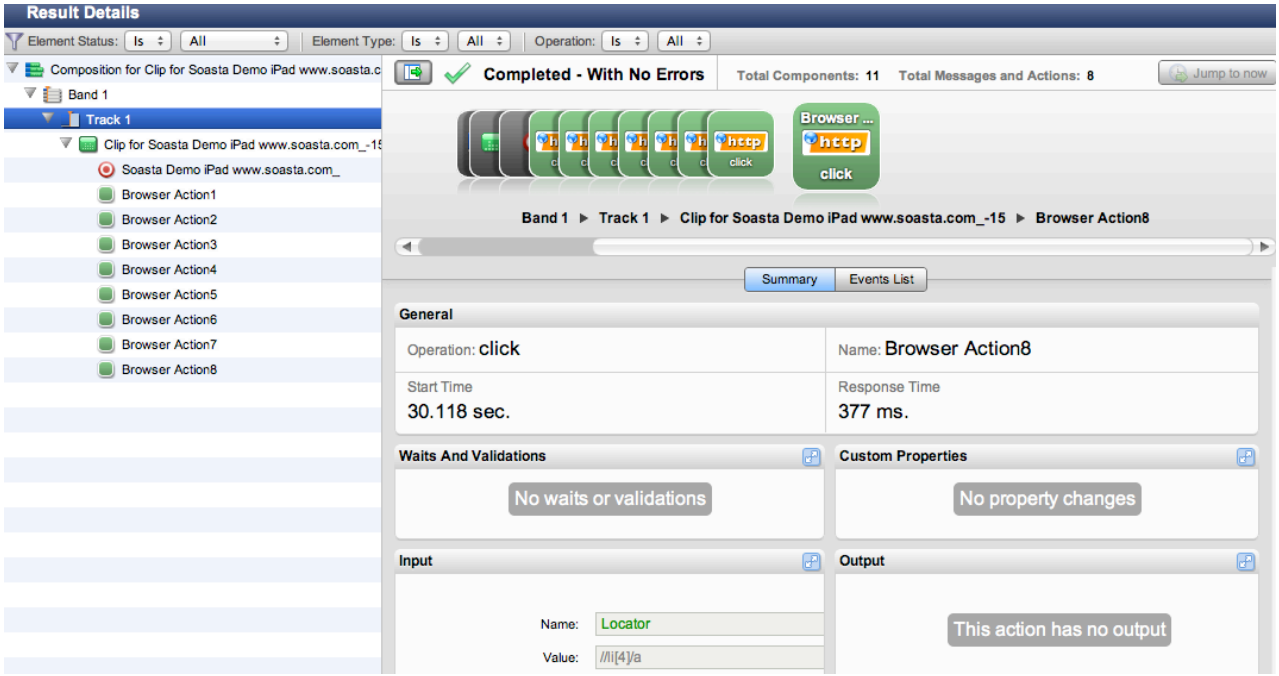
The Result Details dashboard helps to discover the cause of errors in your test, if any. While play continues results are posted in the Composition Editor, Play tab, Result Details widget.



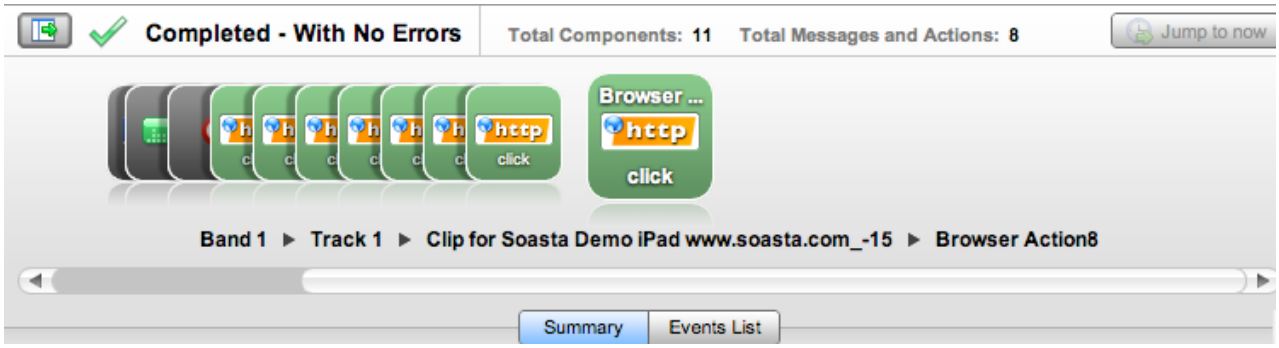
The mobile Browser Actions performed when the clip was created are played back on the device. Click to expand the nodes in the Navigation Tree on the left as they appear.

- Result Details uses a Cover Flow (top panel center) to display the test composition's stream as it occurs.
- This stream is also shown in the Navigation Tree (on the left) as elements are executed. As play continues, the focus is set to the last executed element unless user interaction prevents it. The current container is expanded while the prior containers are closed.
- Clicking an element during play will halt this auto-focus-to-the-last-executed behavior. To resume auto-focus once interrupted, click Jump to Now in the upper right of the dashboard.

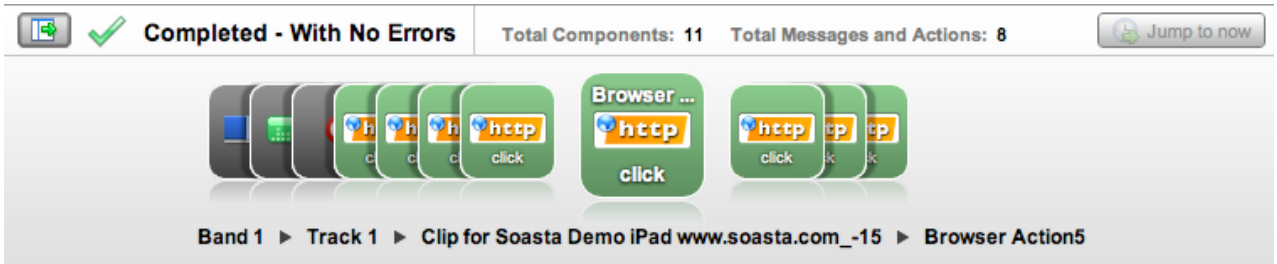
Once play completes, the final results are graded according to the default criteria. For example, the in shot below the test has the status “Completed – With No Errors.”



After the test completes, use the Cover Flow to examine test component details.



- Click any object in the Cover Flow at the top to center it and display its details and play statistics in the panes below.

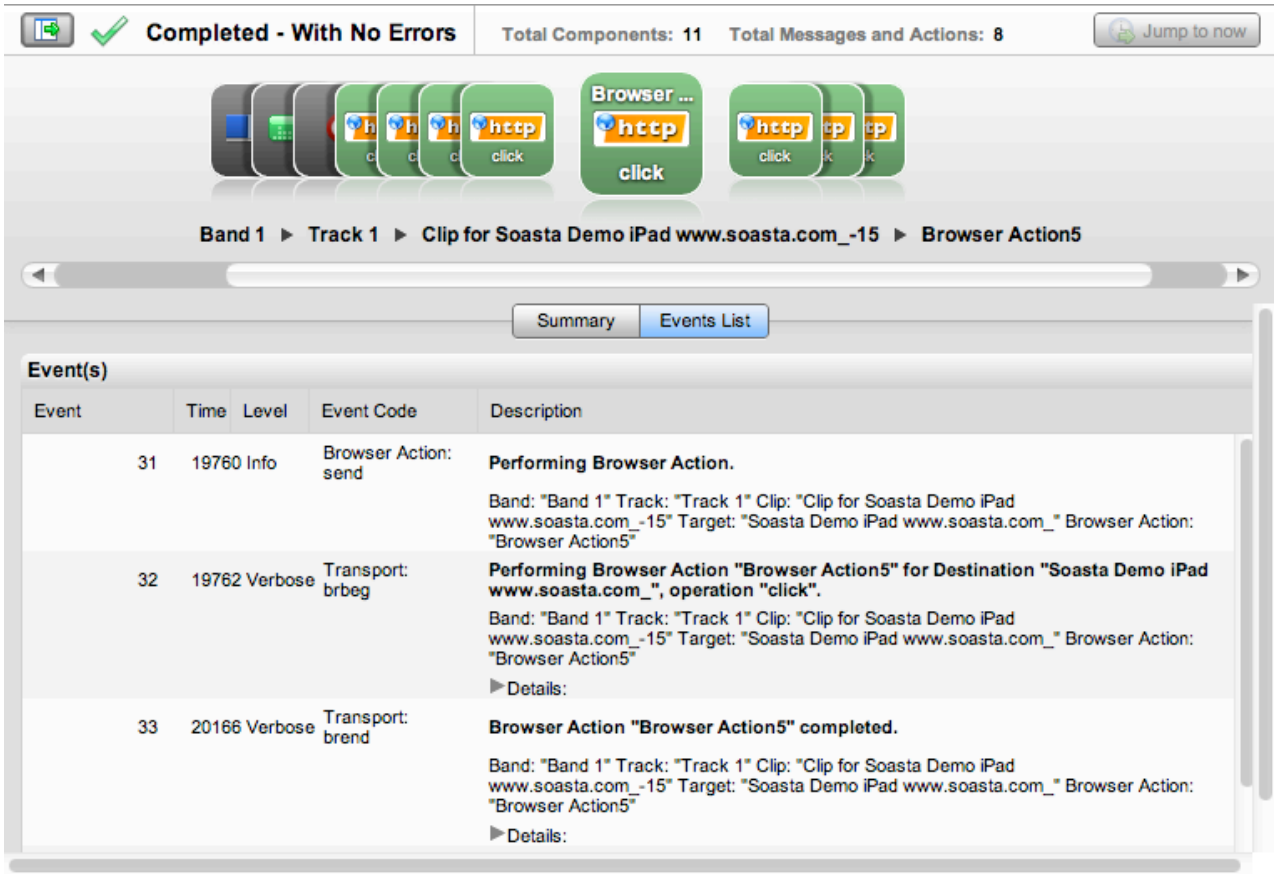


- Use the scrollbar to browse the flow. Select any item to show its low-level details.

Result Details has several methods for navigating through the test results. Click the checkboxes in the cover flow to quickly display messages (or Actions) only (within a single band, track, test clip, or chain).

TIP: Before closing the composition, click the Composition Editor, Edit tab (top right of the Composition Editor) and then click Save on the Composition Editor toolbar. Give the composition a name. For example: "`Composition for <Clip Name>`."

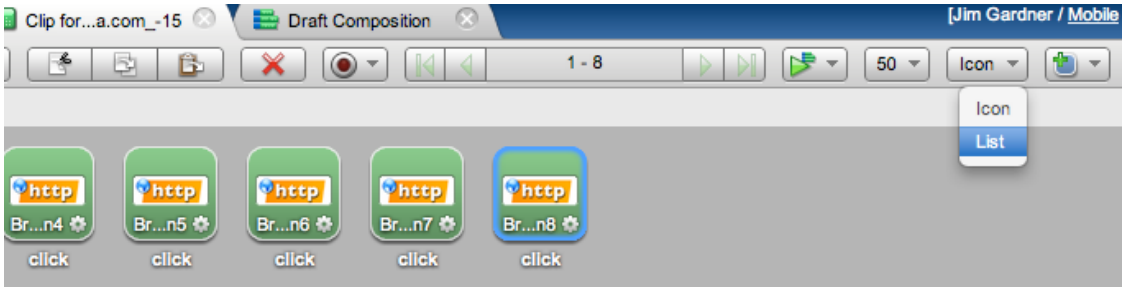
For selected items in either the Cover Flow or the Navigation Tree, additional details appear in the Summary detail sections (lower right) as well as in the Events List tab.



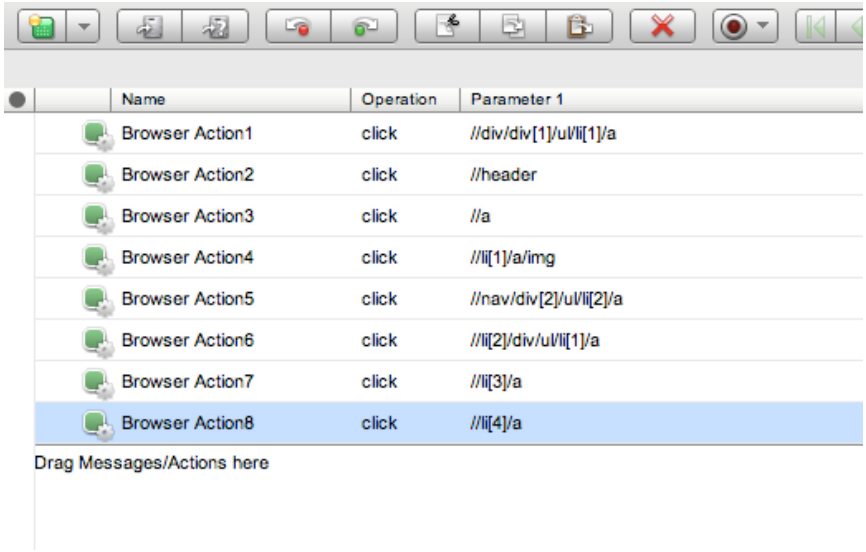
Advanced Clip Editing

Now that we've played this simple test composition successfully, and learned how CloudTest will check the success or failure of a given composition, let's return to the test clip to inspect the clip elements and do some additional parameterization.

- Click the Clip Editor tab if it's still open, or double click the test clip in the Navigation Tree to open it in a new Clip Editor tab.
- Once the Clip Editor tab is in view, click the drop down Icon/List button on the toolbar and then select List.



The List view is useful while clip editing, because it shows all the parameters and their corresponding inputs in one tabular view.

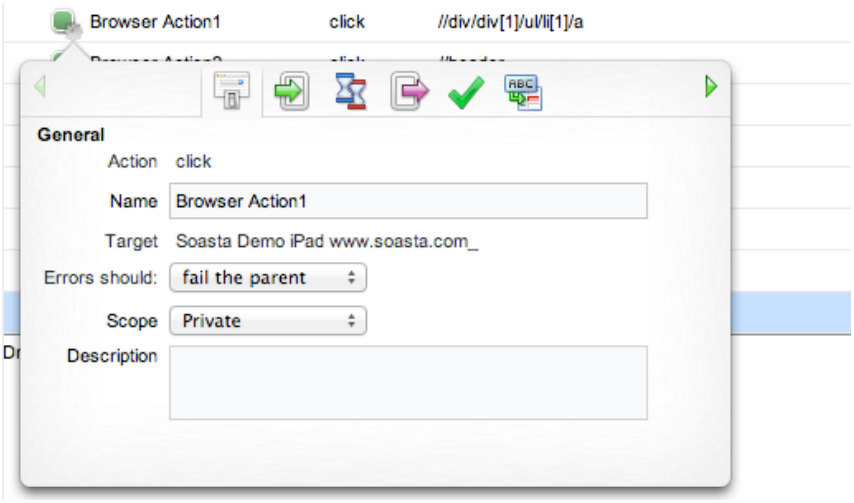


When additional parameters are present they are displayed to the right of the Parameter 1 column.

Inspecting Browser Action Details

Browser Action details can be examined in the Clip Editor workspace by clicking the Gear icon or by opening an item to view it in the lower panel Action Editor. Readers should familiarize themselves with both techniques, keeping in mind that some action details such as those for repeasts and custom properties can only be set in the lower panel.

When you click the Gear icon, the Info Window, General tab appears.



Click the tabs to browse the details for the given action.

In addition to the General tab, the Info Window tabs are Inputs, Waits, Outputs, Validations, and Property Sets.

- **Inputs** (Locator, Scale, Precision, Content Offset)

  Locators are unique characteristics that identify a specific element or object on a mobile device. Locators come in many forms, including links, IDs such as those defined within CSS, and XPath expressions.

- **Waits** (Pre-Action Waits , Post-Action Waits)

  Waits are commands that tell CloudTest not to execute an Action until a condition is met (pre-action waits), or to not continue processing the outputs, validations and property sets of the Action until a condition is met (post-action waits).

- **Outputs**

  Outputs specify what is to be shown in the Result Viewer for a given Action. Typical outputs include “captureScreenshot”, “outputElementText”, and “outputInnerHTML”. A single Action can have an unlimited number of outputs.

- **Validations**

  Validations verify some content or event occurred as expected and have a corresponding Failure Action. Browser Action validations can range from simple true/false conditions to more complex conditions. A single Browser Action can have an unlimited number of validations. Any validation failures will be exposed in the Results Dashboard.

- **Property Sets**

  Property Sets give you the ability to take text or data from the app you are testing and store it in a custom property for use in a subsequent action or message.

SOASTA CloudTest includes three property sets, all of which have relevance for refining and editing a selected Browser Action.

- *Custom Properties*

  Custom Properties are user-defined properties that are available to all clip elements, including Actions. Custom properties can be thought of as backdoors that allow access to portions of the object model more easily.

- *System Properties*

  System Properties are available to all clip elements, including Actions. SOASTA CloudTest defines system properties. For example, a test clip has system properties such as name, repeat timing, label, and more.

- *Global Properties*

  Global properties are defined within the Central > Global Properties List and are “global” within the entire SOASTA CloudTest environment—and can be used across compositions.

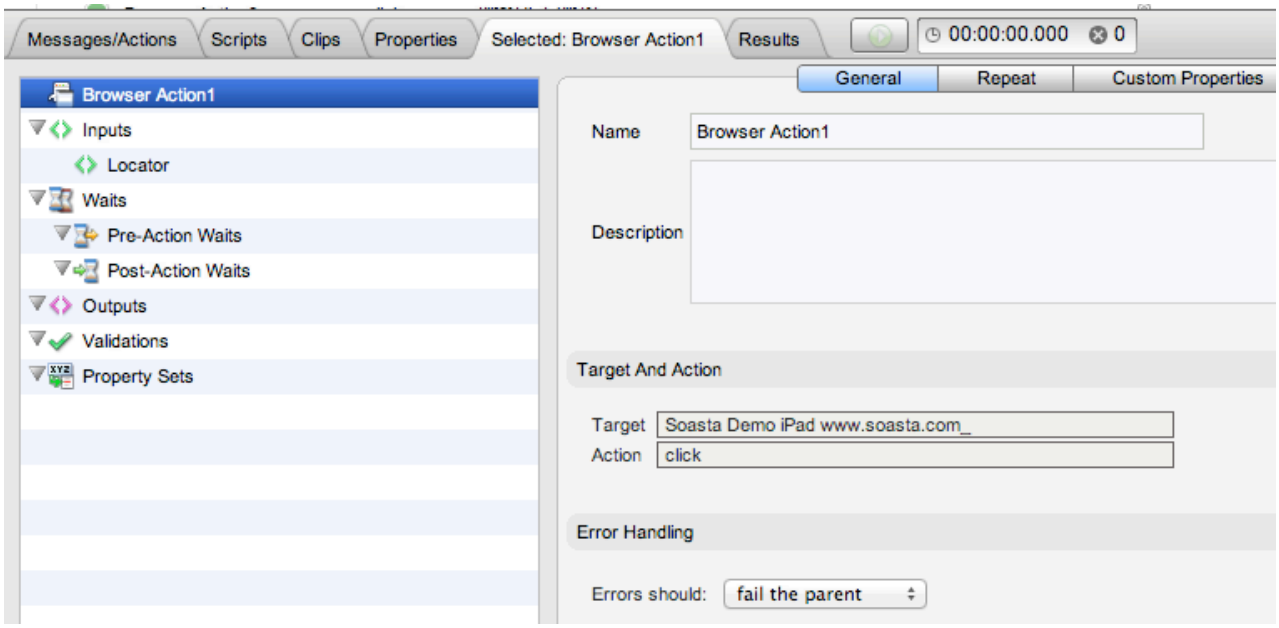
The Info Window disappears when the lower panel is open for a selection—and vice versa. The next section covers action attributes that appear only in the lower panel.

Browser Action Properties

Additional parameters, such as Custom Properties, can be set by double-clicking a browser action to open it in the Clip Editor lower panel. Action level properties are shown in the tree on the left.

1. Select the top-level node in the tree (as shown below)

•General, Repeat, and Custom Properties (for the action only; not for the entire clip) tabs appear on the right. Note that Error Handling here is set to *Errors should fail the parent* by default.



Note in the shot above that when the Browser Action1 (top level node) is selected that the action's Error Handling settings are set to fail the parent by default. When you add accessors such as validations they are children of the action, and can have their own failure actions that differ from those of the parent action.

•In addition to the General tab attributes shown above, the action has Repeat and Custom Properties attributes that can be set by clicking the appropriate tab.

•Other settings, including Waits, Inputs, Outputs, Validations, and Property Sets that were seen in the Info Window steps above can also be set in the lower panel by clicking that node in the tree and then performing the desired action on the right.

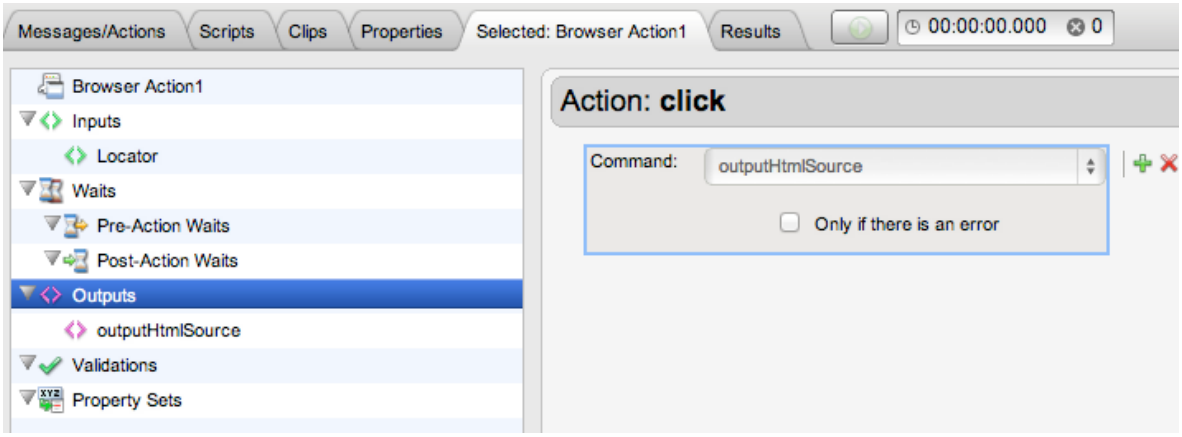
1. In the *Selected: Browser Action1* tab (or for any selected Browser Action), familiarize yourself with the available elements and properties.

Adding Outputs

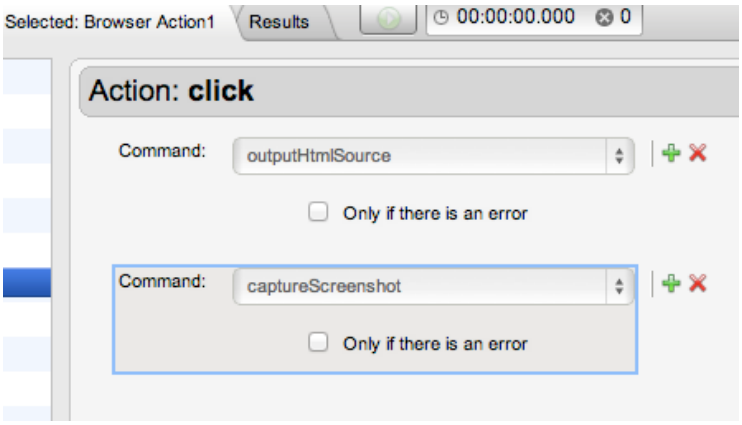
In the following steps, we will add two outputs to the first Browser Action captured from the www.soasta.com site in the recorded clip above—one to output the HTML source in the test clip and another to capture a screenshot during the test.

> **TIP:** If you are the author or programmer for a given site, you are likely already familiar with the source code for your site. However, if you are a tester who is not as familiar with the programming details of a site, getting an output of the source code provides insight into what to test. The following steps can be performed in either the Info Window or, as shown below, in the lower panel.

1. With Browser Action1 still open in the lower panel Action Editor, click Outputs in the list on the left, and then in the *Action: type* field just to the right, click the green Plus sign. A new output form is added with the default output for a browser action, *outputHTMLSource*, shown.



2. Click the Command list drop-down, and then select *captureScreenshot* (it is the top entry in the drop down list).



2. Leave *Only if there is an error* unchecked for now, but note this checkbox for later use in cases where the output is desired only in the event this Browser Action produces an error.
3. Click Save on the Clip Editor toolbar.
4. Return to the Composition Editor tab once again and click Play a second time.

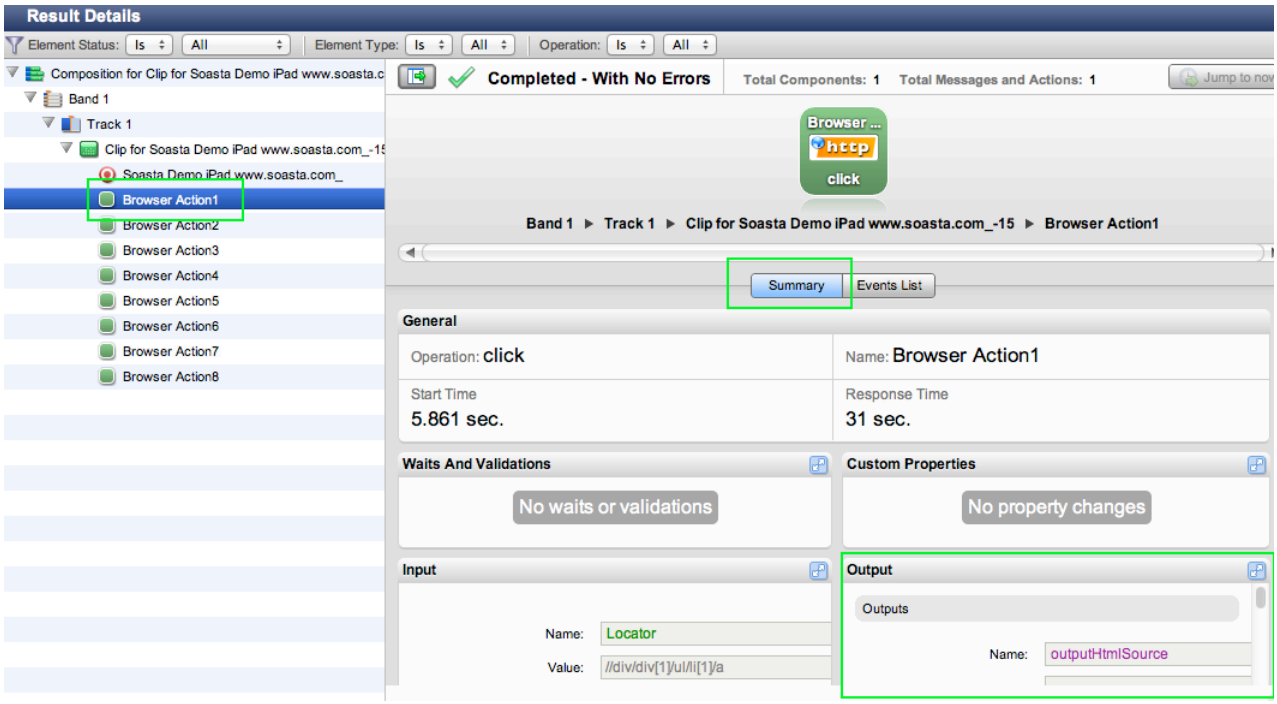
   TIP: If you closed the draft composition without saving it, use Play in Test Composition to create a new draft composition, or if you saved and closed the test composition, click the Central tab, Compositions list and re-open it from there.

In the following section, we'll inspect results for the outputs that were set on Browser Action1 above.

Examining Outputs in Results

In the Result Details dashboard, we will locate and examine the Outputs we set above for Browser Action 1.

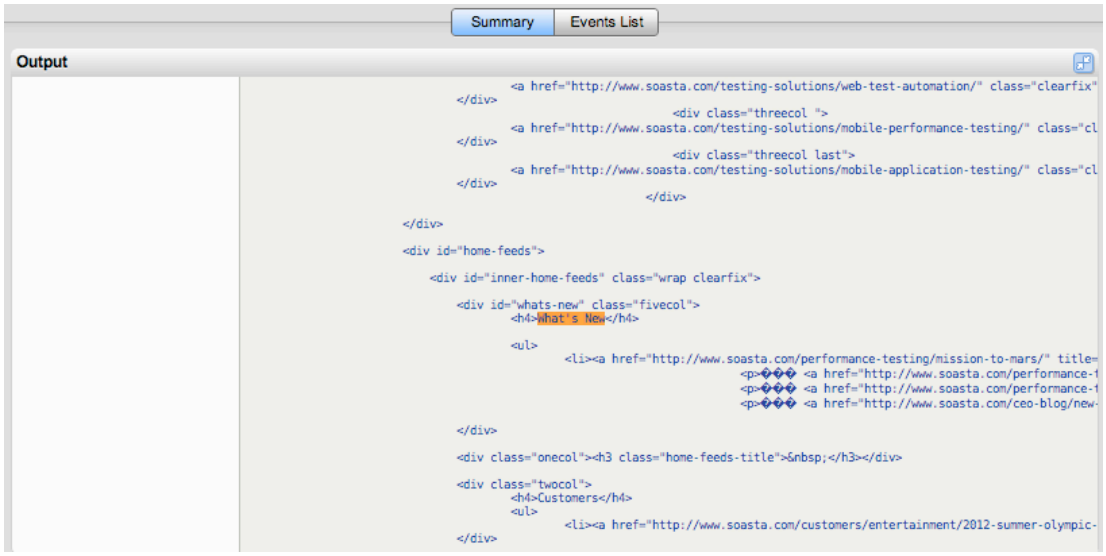
1. Expand the Navigation Tree until Browser Action1 is in display and then select it as shown below.



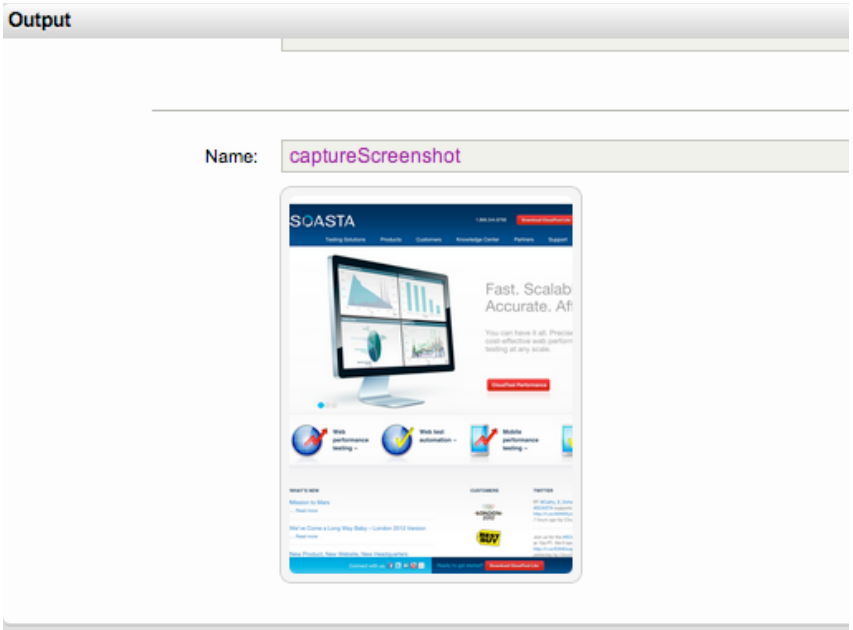
2. Collapse the Monitored Device(s) tab to get a clearer view of the Outputs section.
3. Next, with the Summary tab active, locate and expand the Outputs section by clicking the blue Expand icon in its top right corner.

4. In the maximized Outputs section, locate the What's New text we used when the clip was recorded.

   TIP: Use your browser's search in page feature to locate specific text in HTML source.



5. Next, scroll down and locate the captured screenshot we specified as an output in the prior section.



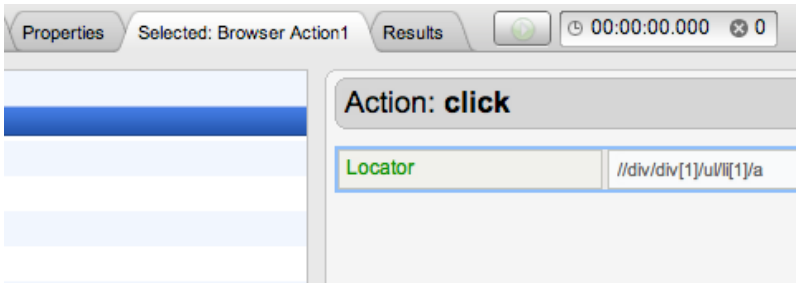
   TIP: You can click the screenshot to pop it out to full view.

In the next section, we'll validate some text in this page.

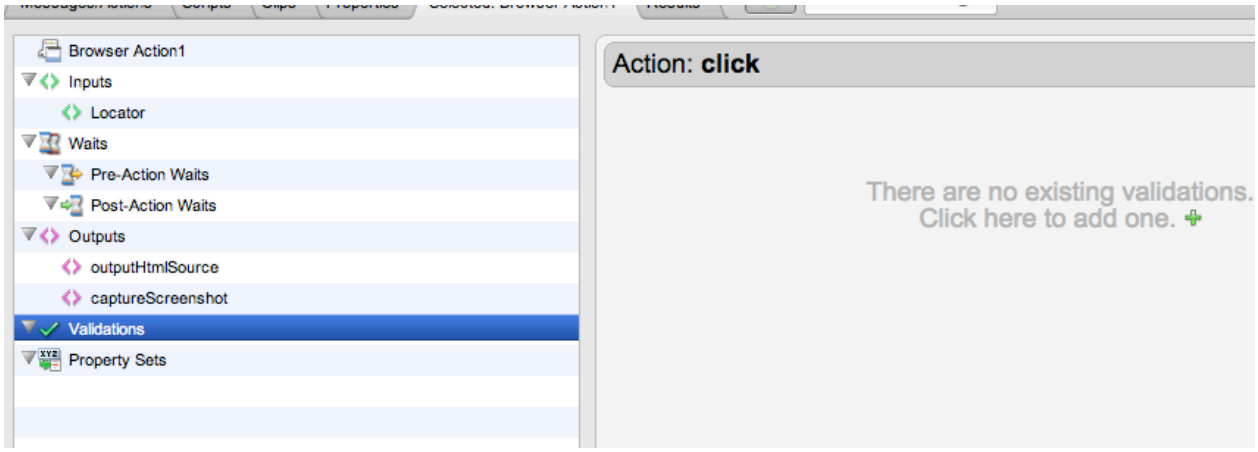
Adding an Element Validation

Next, we will add a validation on Browser Action1. The remaining steps demonstrate how to do simple validation in CloudTest.

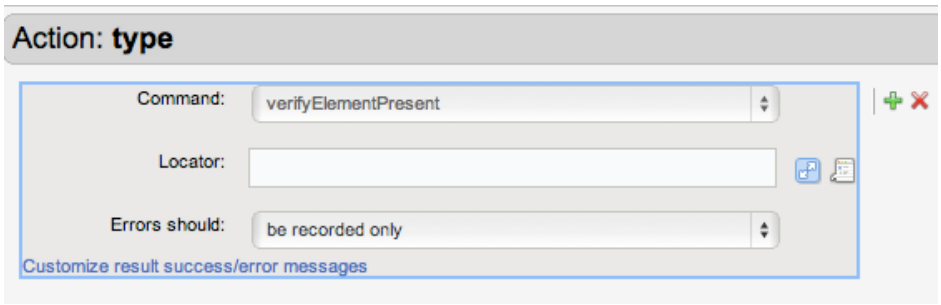
1. If it's not already open in the lower panel, open Browser Action1 now by double-clicking.



2. Click Validations in the list and then click the green Plus sign in the Validations panel on the right.

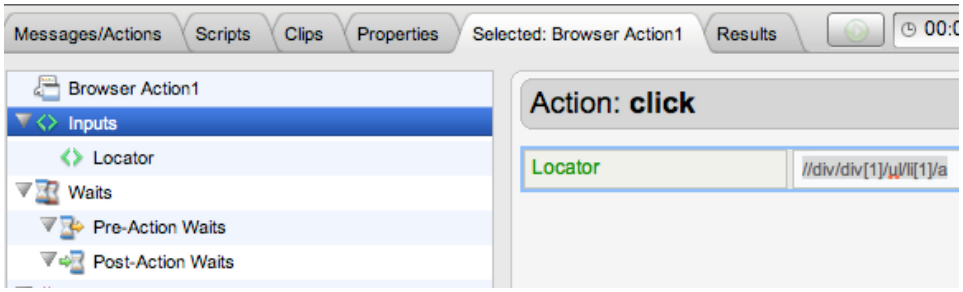


An Action: type form is added to the right panel.

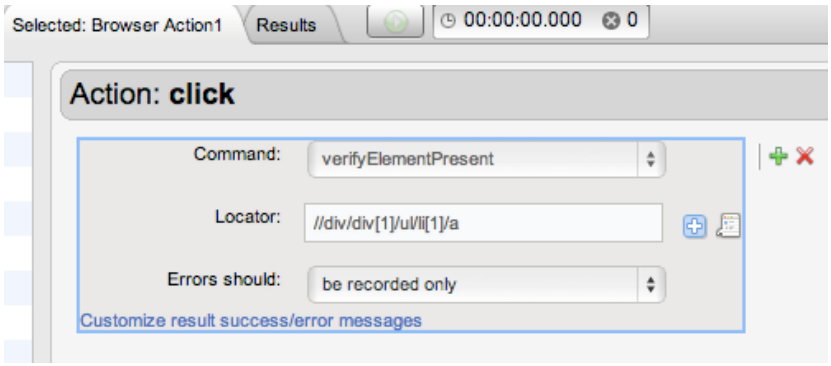


3. Add the Locator that corresponds to Browser Action1.

   This is most easily done by clicking the Input node in the tree on the left and copying the Locator as it was captured.



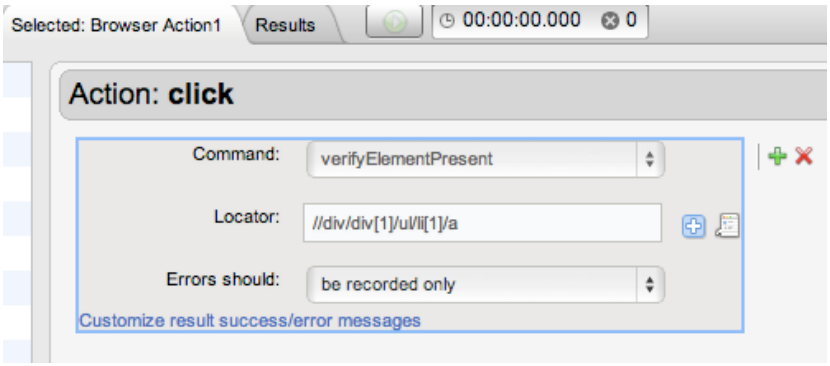
After which, the Locator can be pasted into the Validation form, Locator field.



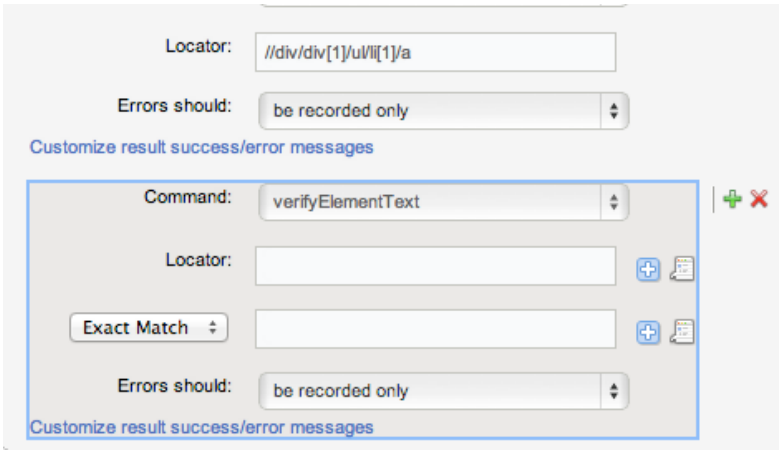
Adding a Text Validation

Next, we will add a text validation on Browser Action1.

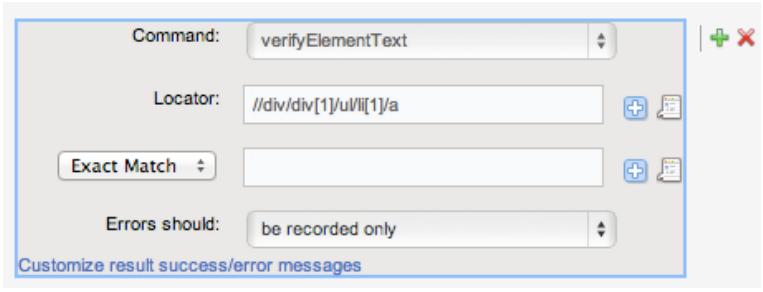
4. Click the green Plus icon to the right of the previous validation that was added above.



5. Click the Command ldrop-down and select *verifyElementText*. This Command verifies that the specified text is in the rendered field.

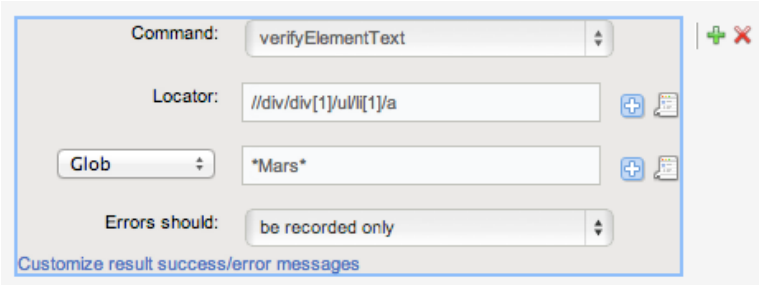


6. In the Locator field for verifyElementText, enter the Locator for the browser action. This can be copied from the Locator field in the validation above.



7. Click the Match field drop-down, and select *Glob* and enter the match text. For example, when the clip was recorded the first link in the What's New field was *Mission to Mars* (this link led to an article about SOASTA's work with NASA). The

text we are validating in this case can be any link text. Given that example, we entered **Mars** as our Glob text.

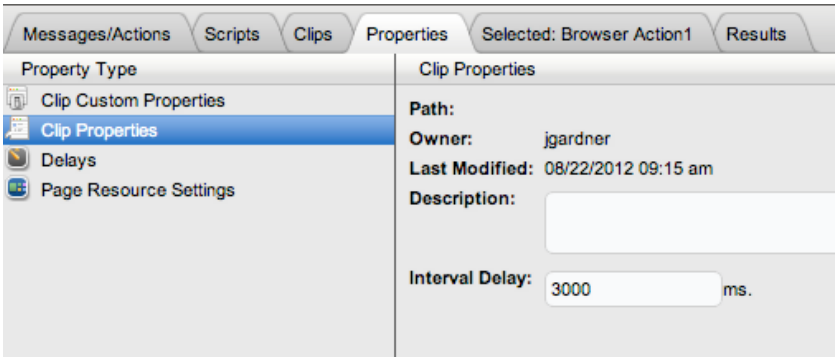


8. Click Save on the Clip Editor toolbar.

Adding an Interval Delay between Each Action

Out test clip can succeed as is when played in a test composition. In some cases, however, we may want to improve the playback on the device to better show the pace of the test as it plays. In the following steps, we will add an interval delay to the test clip. This type of delay will stretch out the time between all the recorded app actions.

1. Click the Properties tab in the minimized sub-panel and then select the Clip tab at the top of the pane (the Clip tab may already be visible if properties are already open from the prior exercise).
2. In the Property Type list, click Clip Properties.
3. In the Clip Properties panel on the right, enter an Interval Delay in the given field. For example, *2000* ms. Entering *3000* adds a three second gap between each app action in the given test clip. This is in addition to any waits that we may also add between specific actions (or which TouchTest technology may build into some clips).

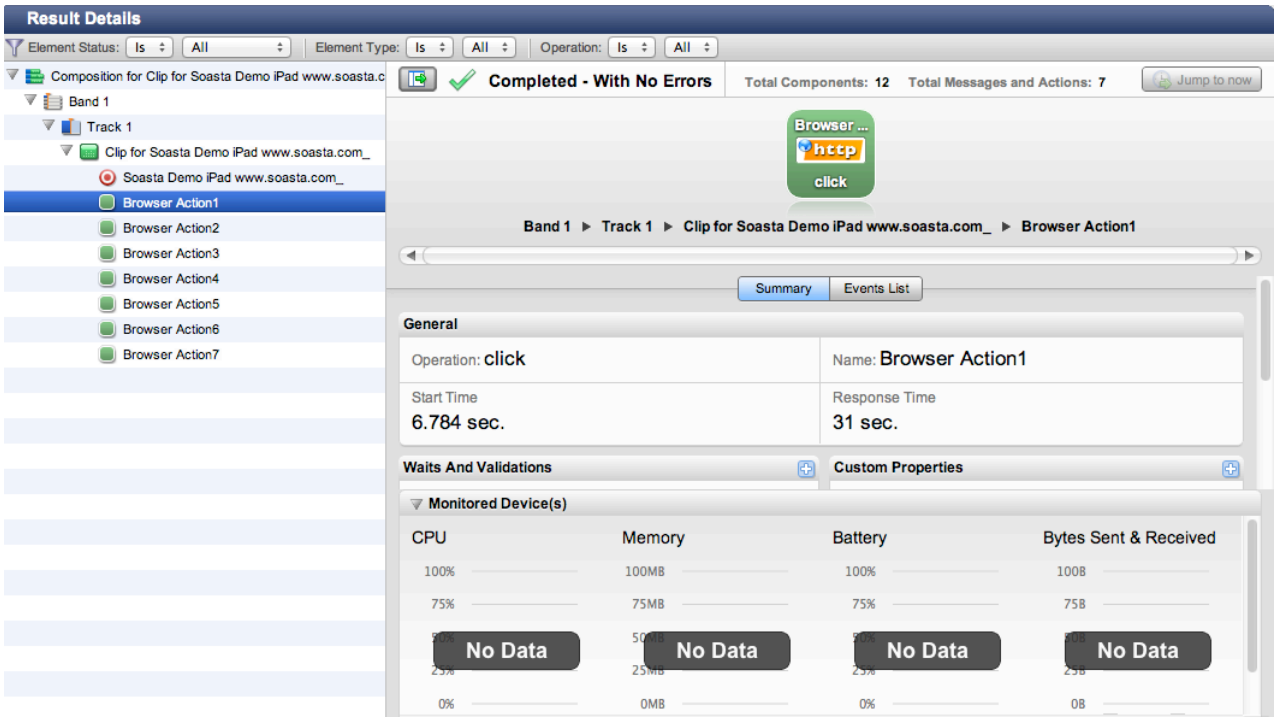


4. Click Save on the Clip Editor toolbar.

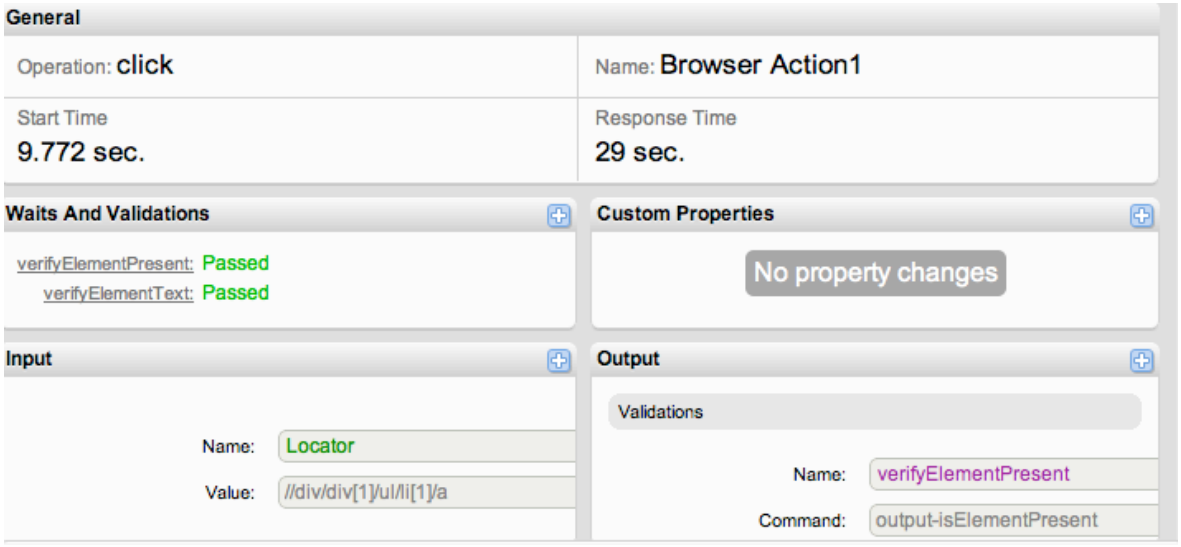
Analyzing Validations in Results

Now that we've added validations on a page element and text, we will learn how to examine those validations on their merits. In the best case scenario, the parameters added in Advanced Clip Editing work without a hitch. We can easily verify the status of our parametere in Result Details.

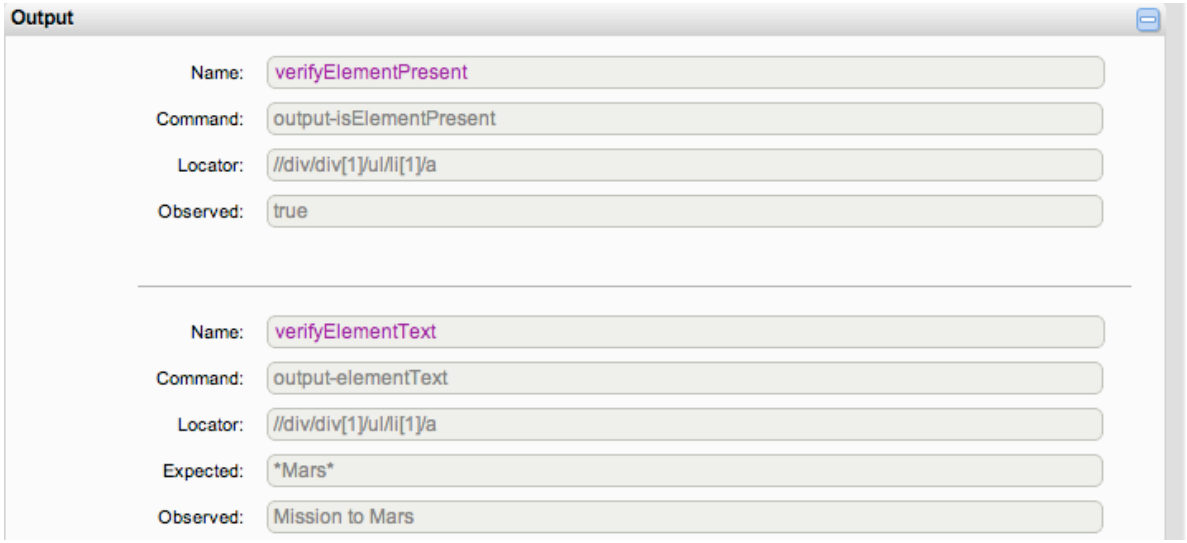
1. Expand the Navigation Tree until Browser Action1 is in display and then select it as shown below.



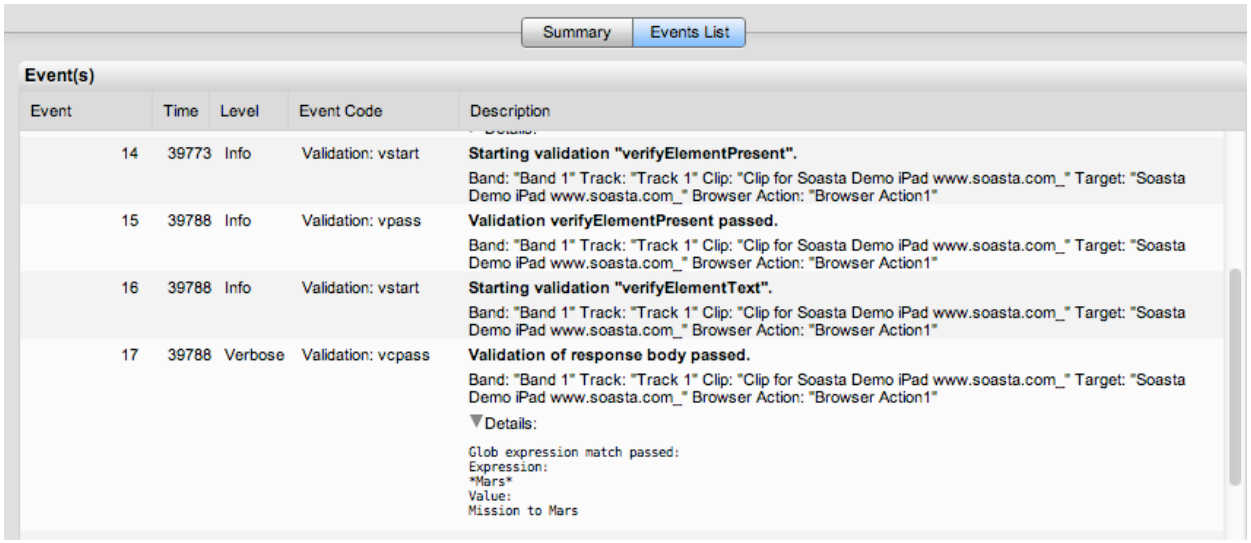
2. Collapse the Monitored Device(s) panel to get a better view of the Waits and Validations and Output panels.



5. To drill into details of the validation, click it in the Waits and Validations panel. This focuses the Outputs panel onto the given validation.
6. Click the Maximize icon (top right of the Outputs panel) to get a wider view.



7. Click the Events list tab for the given selection to view action-related events, including validations. Click the Details arrow to inspect any event's details.



SOASTA, Inc.
444 Castro St.
Mountain View, CA 94041
866.344.8766
http://www.soasta.com