

jQuery Selector Optimizer

jQuery should be used for edge cases only

We recommend using jQuery as little as possible. Web applications are constantly changing and small updates to the UI and page structure can break any jQuery used.

WalkMe's out of the box element selection adapts to changes in the underlying application without the use of jQuery.

Brief Overview

With most WalkMe items it is possible to freely input jQuery selectors to select elements. The jQuery library has existed for quite a while, but its syntax and way of writing selectors has changed over time.

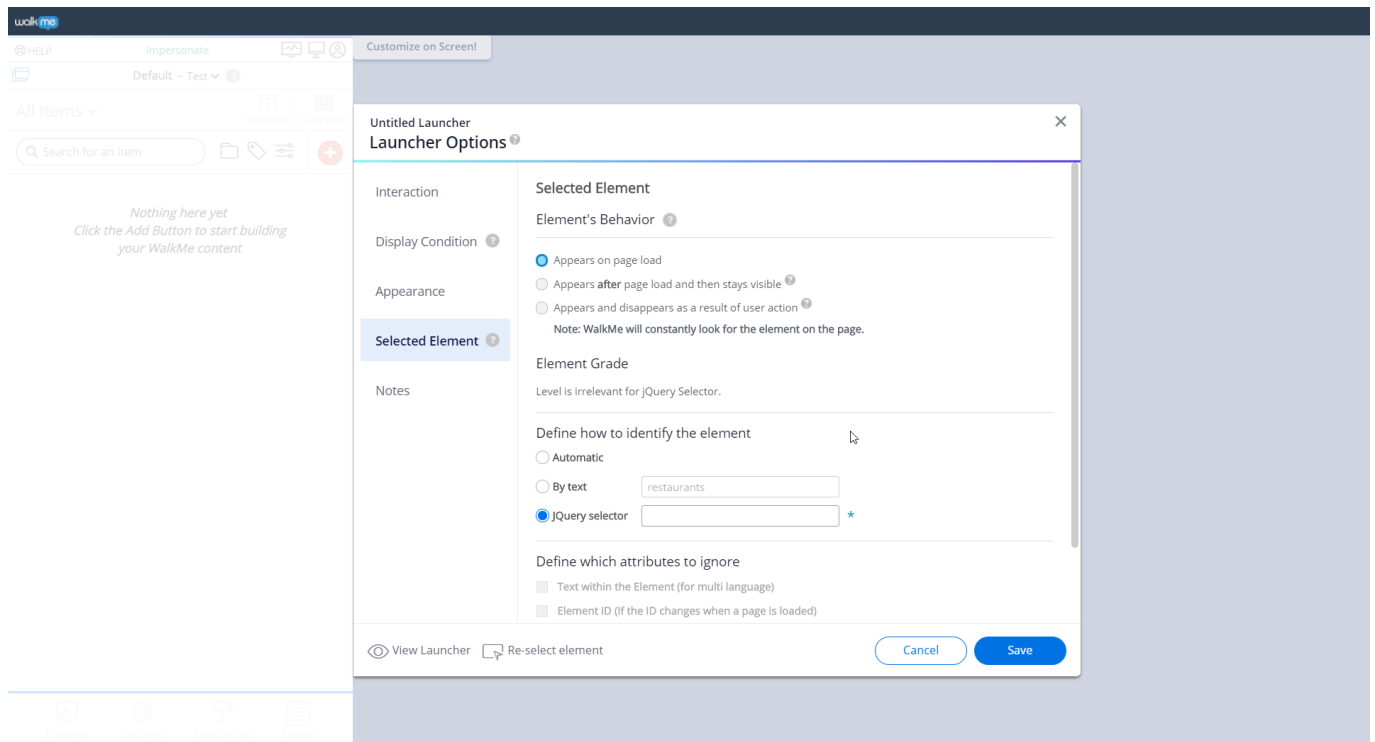
For users who are not familiar with the latest syntax and use an old-fashioned way of writing jQuery selectors, this can sometimes cause a longer evaluation time. This becomes a problem when users have a lot of content with old-fashioned selectors.

We want to improve WalkMe's performance of finding elements on the page and help you write more efficient selectors.

The '**jQuery Optimizer**' detects if a specific selector that you have written can be rewritten with a more modern syntax that would evaluate faster. When it does, it automatically updates the value in the field with the optimized selector, leaving the option to revert the change manually.

jQuery Cheatsheet


For jQuery tips and examples, please refer to our [jQuery Cheatsheet](#)



How It Works

This feature is automatically available for any deployable that has the **“Define how to identify the element”** option within the settings of that particular deployable:

Define how to identify the element

☐ Automatic
☐ By text
☒ JQuery selector 

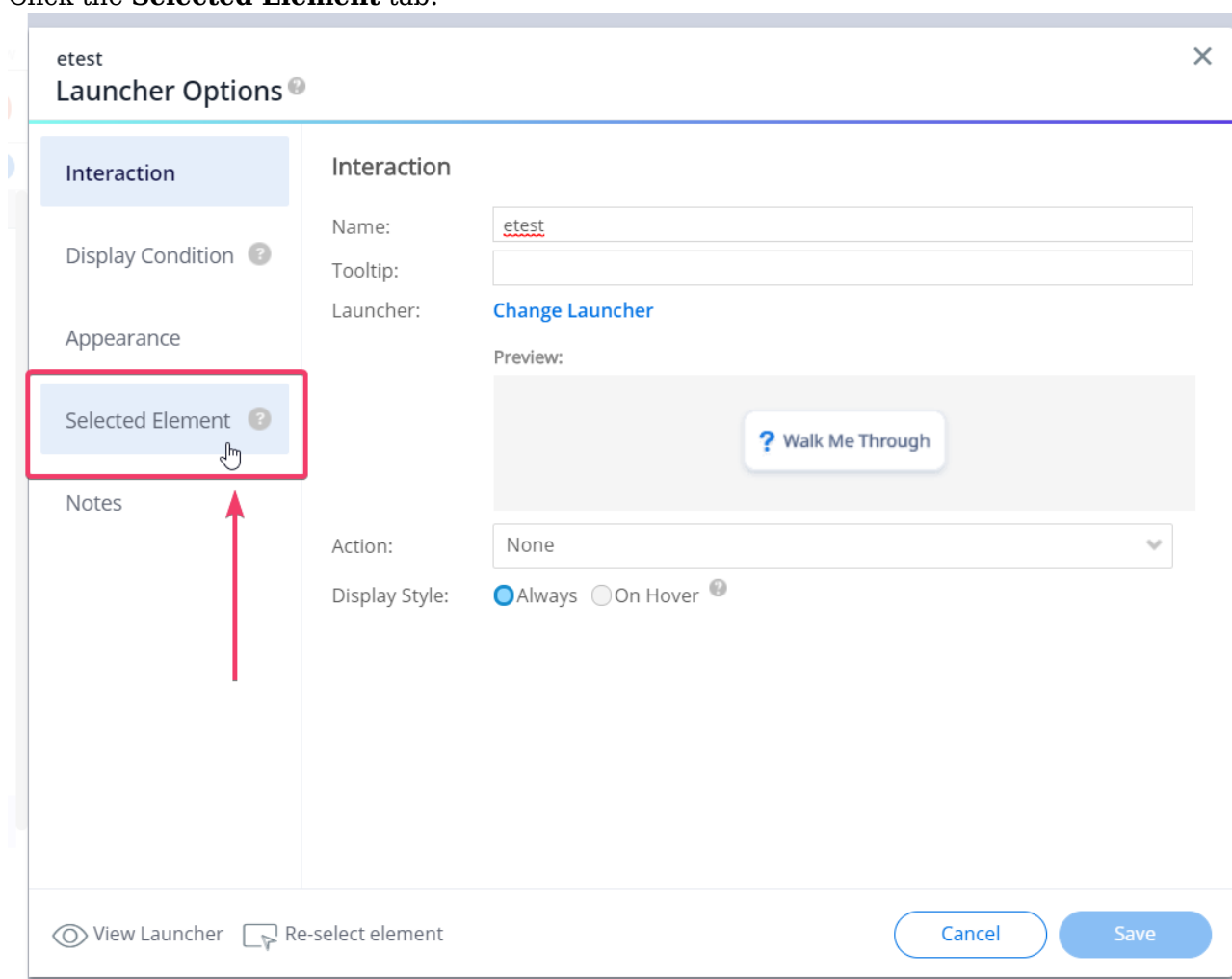
Optimization examples

Before optimization	After Optimization
[class="foo"]	.foo
[class="foo bar"]	.foo.bar

[id="foo"]	#foo
[class="foo bar"] [id="foo"]	.foo #bar
[class="foo"].bar:not([class="foo-bar"])	.foo.bar:not(.foo-bar)
[class="foo"][class="bar"]:not([class="foo-bar"])	.foo.bar:not(.foo-bar)

Steps for using the JQuery Optimizer

1. Open any deployable with a **Selected Element**.
2. Click the **Selected Element** tab:




The screenshot shows the 'Launcher Options' dialog box for an element named 'etest'. The 'Selected Element' tab is selected and highlighted with a red box. A red arrow points to this tab. The dialog is divided into two main sections: a left sidebar with tabs (Interaction, Display Condition, Appearance, Selected Element, Notes) and a right main area. The 'Selected Element' tab is active, showing fields for Name (etest), Tooltip, Launcher (Change Launcher), and Action (None). There is also a Preview area showing a 'Walk Me Through' button and a Display Style section with 'Always' selected. At the bottom, there are buttons for 'View Launcher', 'Re-select element', 'Cancel', and 'Save'.

3. Select **jQuery selector** in the "Define how to identify the element" section:

Selected Element ?

Notes

☒ Appears and disappears as a result of user action 

Note: WalkMe will constantly look for the element on the page.

Element Grade

Level is irrelevant for jQuery Selector.

Define how to identify the element

☐ Automatic

☐ By text

☒ jQuery selector
 *

4. **Fill value** written in an old-fashioned way, for example: [class="foo bar"], and **click out** of the input field:

Selected Element ?

Notes

Define how to identify the element

☐ Automatic

☐ By text

☒ jQuery selector
 *

5. Watch the **magic** of your selector being optimized!