Target Builder

Target Builder allows you to create custom targets from the data you already have about your customers without writing any code. You can create these targets, used in the WHO part of an experience, from cookie values, HTML elements, JavaScript variables, query parameters, and more.

Use Target Builder to create a custom segment that falls outside the default options available in the Monetate platform. For example, you can create a custom target for site visitors who have a component, such as a cookie or JavaScript variable, with a specific value set using your CRM data. Any target created in this tool you can reuse in an unlimited number of experiences.

Cookie-Based Targets

You can use any of your site's cookies or information contained within these cookies to create custom targets. A potential use case for cookie-based targets is to target based on a cookie for a custom segment set by your analytics tool.


If you build a cookie-based target but it doesn't work, understand that the cookie must have a domain attribute set, which then sets the cookie to hostOnly. This configuration means that the cookie value can only be read by requests from servers within the host domain (yourdomain.com). The cookie target evaluations are made by monetate.net servers, which is why the cookie is not working as expected.

To work around this behavior, you must set a domain attribute for the cookie when it is written by your script. Doing so removes the hostOnly flag from the cookie and thus allow the platform to evaluate it for targeting purposes.

HTML Element–Based Targets

You can create a target based on the presence of specific DOM elements or the contents of DOM elements on your site.
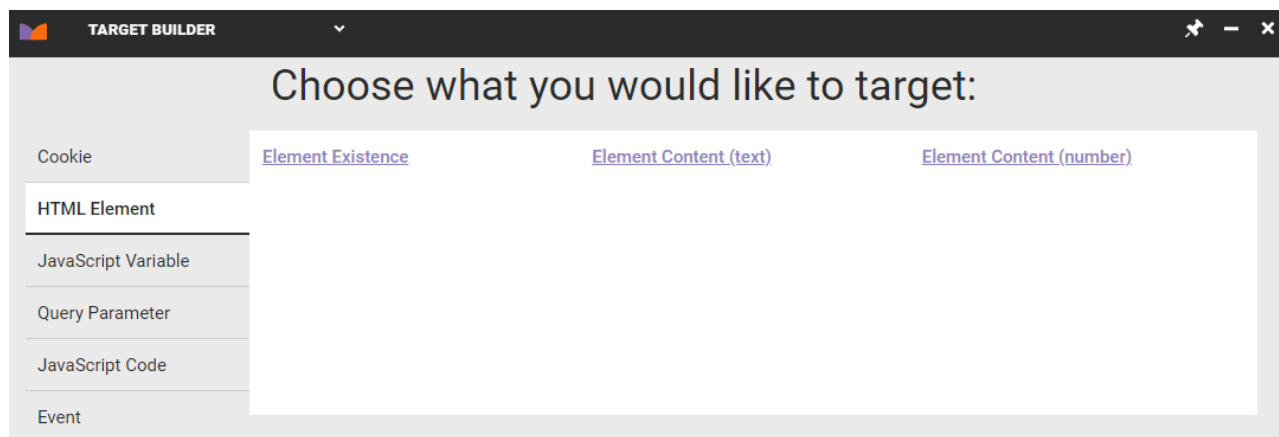
A potential use case for HTML element–based targets is to target visitors who land on a page that contains the words *out of stock* within a specified page element.

JavaScript Variable-Based Targets

You can use any of your site's JavaScript variables or information contained within these variables to create custom targets. For example, if you have a JavaScript variable that stores when a user is logged into a website, you can use that information to create a custom target.
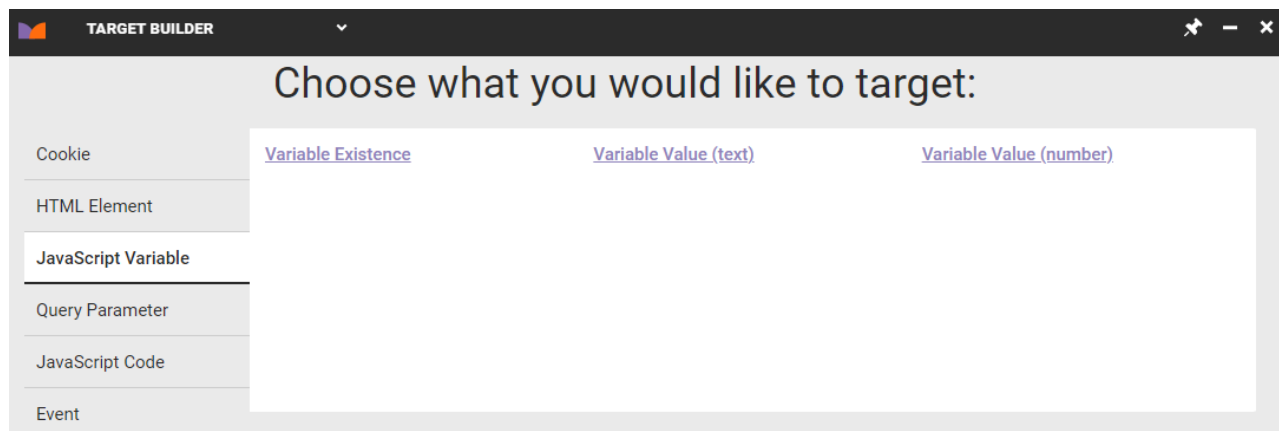

Here are a few use cases for in-browser JavaScript variable targets:

- **navigator.userAgent** — Target in-store kiosks or browsers not available in the WHO via their user agent string, such as

  Mozilla/5.0 (Windows NT 10.0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/42.0.2311.135 Safari/537.36 Edge/12.10136

- **document.referrer**: Target users from a particular URL within your website, such as your site's cart at http://example.com/cart/
- **document.title** — Target users that see a webpage with specific title text as shown in the browser

Query Parameter-Based Targets

You can use any inbound query parameter or query parameter value to create custom targets. For example, you

can target based on a query parameter when a customer arrives on your site from an email clickthrough.
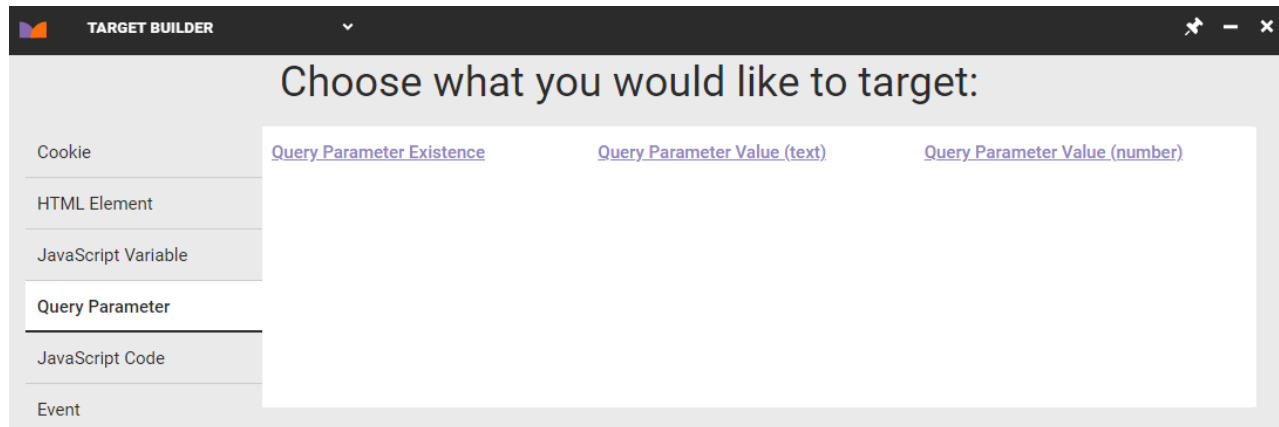
An experience based on a query parameter target only shows when the parameter is present in the URL. If the URL contains the parameter `?source=email`, then enter `source` in the **Query Parameter Name** text field and `email` in the **Query Parameter Value** text field when building the target.

> ❗ Do not include the equals sign (=), the question mark (?), or the ampersand (&) in the **Query Parameter Name** and the **Query Parameter Value** text fields.

JavaScript Code–Based Targets

A target you build with this option runs the specified JavaScript code that you enter, and then if the code returns a truthy value, it counts the target as having matched. The code you supply must therefore include `return variable;` or some other instruction to return the value.

This target type is useful if you need to mitigate timing issues and add polling to a JavaScript target.

> ❗ Do not use this type of target as a means to build out a completely new function or evaluation.
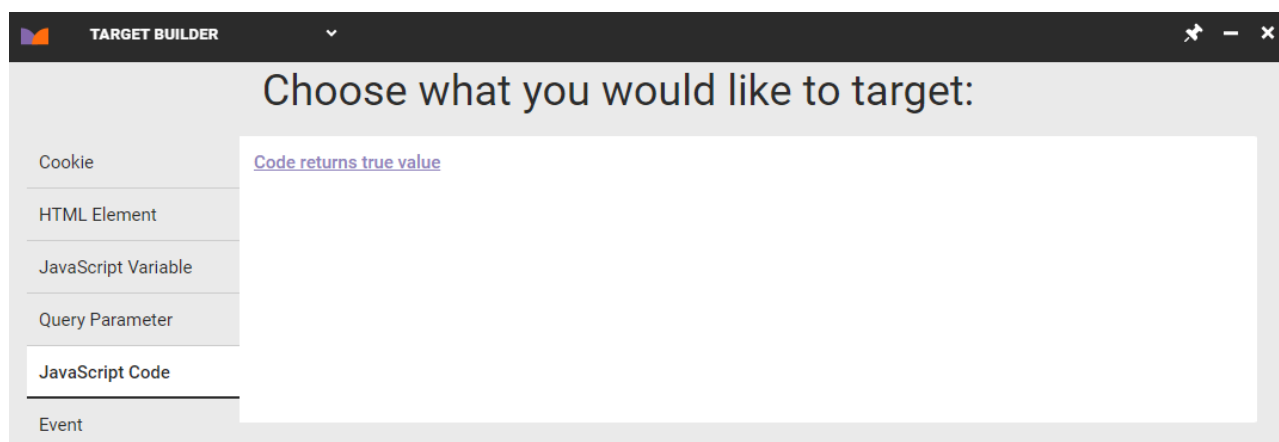

If there's no immediate match, then the target's code continues to run, waiting for a truthy value to be returned,

for the length of time (in milliseconds) that you specify. This interval repeats until either there's a match or the total time (in milliseconds) that you specify has elapsed.

If the code return a true value before the target reaches the timeout threshold, then a retrack fires with the exact same data as the previous track with the target matched added but without a page view so that the page view counts aren't artificially inflated.

Event-Based Targets

Use the **Target DOM Event** option on the **Event** tab to create a target that listens for a specific event, or type of interaction, related to an element that's taken either by a site visitor or the visitor's browser.
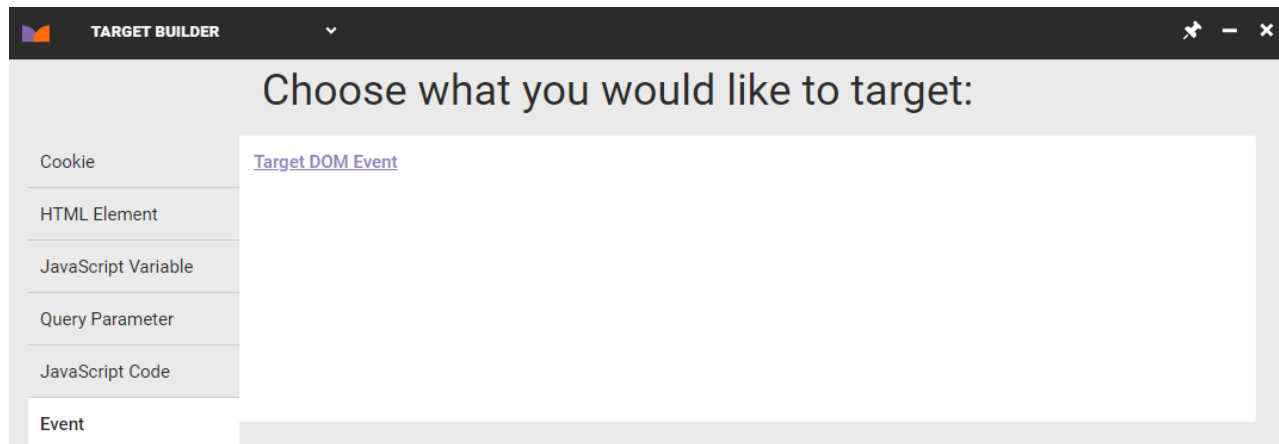
The events can include the following:

- Mouse events, such as click, mouseover, mousedown, mouseup, and mouseout
- Form events, such as submit, change, and focus
- Keyboard events, such as keypress, keydown, and keyup
- Window events, such as scroll, resize, and load

See Event reference from MDN Web Docs for more information about events.