

# Detach Node Action

The Detach Node action entirely destroys a node or element, including any child elements it contains. The node is completely removed from the DOM (Document Object Model, the rendered webpage content).



Detach Node is entirely unlike a Hide action, which leverages inline CSS to hide an element rather than entirely removing it. Detaching an element from the DOM may have unintended consequences, especially if that element was necessary to a JavaScript function. A Hide action keeps any functionality intact because the element is merely invisible to the visitor but remains intact in the DOM. Detach Node utterly destroys the element and all bindings to it.

Due to the highly destructive nature of this action, you should only use it under certain circumstances and with extreme caution and very thorough testing.

## Element Selector (Required)

Enter your element selector in the **CSS3 Selector** field. This selector determines where the image creative is inserted. For additional information on selectors, refer to [Element Selectors](#).

## Select All Elements Toggle

The **Select All Elements** toggle allows you to apply the action to the first occurrence of the CSS3 selector or all occurrences of the selector (if multiple exist).

Selecting **No** applies the action to the first occurrence. Selecting **Yes** applies the action to all occurrences.

## Re-check for Elements Toggle

Use the **Re-check for Elements** toggle to enable or disable polling. If you enable polling, Monetate continues to check for the selected element even if it does not initially appear on the page. This is a good option if you have certain elements that don't fire right when the page loads.

If you disable polling, Monetate only checks for the element once when your page loads.

## How and When to Effectively Use Detach Node

### Hiding Element(s) That Can't Be Hidden

There are some instances where an element(s) simply cannot be hidden with CSS, even when used inline via a Hide action. A prime example is an `<option>` in a `<select>` menu. There is no easy way to hide or remove a single option from within a list using traditional methods. The element itself must be moved or removed from its parent to properly hide it from the visitor.

## Other Uses with Dynamic Content and Retracks

Detach Node is an action that, when properly used, enables other Monetate actions to run as intended when dynamic content and retracks are involved by destroying the Monetate-inserted content from the initial page track. This allows it to be re-inserted afresh on each subsequent retrack.

While Monetate does employ several features in the platform that prevents duplicate actions from running or conflicting experiences that target the same code on page, it is still possible for duplicate content to be added unintentionally during a retrack. If an action is inserted and that action has no slotting, it may be inserted again on each subsequent retrack. Most actions contain action slotting by default, but this remains a possibility.

There are other circumstances in which Monetate safeguards that prevent duplicate content need to be actively circumvented. When you run an action that targets multiple elements on a page and that action has slotting, it's prevented from running on each subsequent retrack by its slot, even if a retrack on dynamic content has created more element(s) that action should target. In this scenario, you must leverage a Detach Node action to remove any trace of the action originally fired on the first track so that it may run cleanly on the second retrack on the new element(s) as well as the originals.

## How to Properly Configure a Detach Node Action or Experience

Configure Detach Node locally within a single experience by leveraging action priority or globally by leveraging experience priority.

### Local Use

For a single element in a single experience that has conflicts with slots and retracks, it is best to use Detach Node within that experience. Detach Node must run as the first action within the experience, and you must never enable the **Re-check for Elements** toggle in the action.

### Global Use

For some websites running frameworks such as Angular or React, there are significantly more dynamic elements on a page and subsequently, more frequent retracks are needed by Monetate. While it is not advisable nor intended to be a long-term solution, it is possible to set up a single experience running at the highest priority to all audiences (a no-control experience) that removes all Monetate-inserted content from the DOM.

This type of experience enables all other experiences running at a lower priority to deliver actions on every page track as if it were an initial page load. Detach Node must run by itself in an experience, and that experience must remain the highest priority. Additionally, you must never enable the **Re-check for Elements** toggle in a Detach Node action.