Developer Hooks and Filters Reference for Add to Cart Button Labels

Developers · https://wpfactory.com/docs/add-to-cart-button-labels-hooks-filters/

Developer hooks and filters reference

This reference lists the filters exposed by Change Add to Cart Button Text for WooCommerce and the WooCommerce filters it hooks into, so you can extend or override its behavior in code. All examples go in your child theme's `functions.php` or a small custom plugin.

WooCommerce filters the plugin uses

The plugin sets the button text by hooking WooCommerce's own button-text filters at priority `PHP_INT_MAX`, so it runs after most other code:

Filter	Applies to
`woocommerce_product_single_add_to_cart_text`	The button on the single product page.
`woocommerce_product_add_to_cart_text`	The button on shop and archive listings.

To have the final say over the text after the plugin runs, hook these at an even higher priority is not possible (the plugin already uses the maximum), so add your callback and return your value; because the plugin resolves the label first, the simplest override is to hook the plugin's own filters below.

Plugin filters

These filters are registered by the Pro class and control the Pro gating. They are the intended extension points.

Filter	Purpose	Arguments

`alg_wc_add_to_cart_button_labels_per_category`	Number of category groups to render and evaluate.	`$count` (default 1 on free)
`alg_wc_add_to_cart_button_labels_per_tag`	Number of tag groups.	`$count`
`alg_wc_add_to_cart_button_labels_per_user_role`	Number of user role groups.	`$count`
`alg_wc_add_to_cart_button_labels_per_user`	Number of individual user groups.	`$count`
`alg_wc_add_to_cart_button_labels_on_sale`	Whether the current product counts as "on sale" for the On sale condition.	`$value, $product`
`alg_wc_add_to_cart_button_labels_settings`	Gate filter: controls the read-only state and group-count caps that mark features as Pro.	`$value, $type, $atts`

Example: change how "on sale" is decided

```
add_filter( 'alg_wc_add_to_cart_button_labels_on_sale', function ( $is_on_sale, $product ) {
    // Treat products in the "clearance" category as on sale.
    if ( has_term( 'clearance', 'product_cat', $product->get_id() ) ) {
        return true;
    }
    return $is_on_sale;
}, 10, 2 );
```

Example: raise the number of category groups

```
add_filter( 'alg_wc_add_to_cart_button_labels_per_category', function ( $count ) {
    return 5; // render and evaluate five category groups
} );
```

The group-count filters are what the Pro version uses to lift the free one-group cap. On the free version the caps also apply to the admin field, so raising the count in code is intended for Pro-licensed extensions.

Note on internal naming



WPFACTORY ACCOUNT

See every feature and the pricing on the WPFactory account plugin page.

View plugin →

Options and hooks use the historical `alg_wc_atcbl_` and `alg_wc_add_to_cart_button_labels_` prefixes. The button-label shortcodes use the `alg_wc_atcbl_` prefix (for example `[alg_wc_atcbl_product_price]`). These are current, not deprecated.

Need a hand?

For integration questions, our team answers through the plugin support page.

Related articles

- Add to Cart Button Label Shortcodes Reference
- How to Set Add to Cart Button Labels by Product Tag
- Set Add to Cart Button Labels per Language (WPML and Polylang)
- Default Add to Cart Button Labels by Product Type and Condition
- Free vs Pro: Add to Cart Button Labels Feature Comparison