

AffStackDev Free Gifts & BOGO Complete Customer Reference

AffStackDev | Release 1.0.10 | Documentation revision 2026-10-22.1

Edition date: October 22, 2026.

Audience: Store owners, administrators and agency staff.

About this reference

AffStackDev Free Gifts & BOGO lets you create automatic and customer-choice gifts, choose qualifying purchases and eligible rewards, configure conditions and allowances, display current offers and understand their cart and order results. This reference covers every supported setup and operation for merchants, administrators and agency staff, including shopper selection and recovery when an offer cannot proceed.

Use the contents and feature index to find a setting, requirement or task. The companion Master Technical Reference supplies exact fields, calculations, permissions, hooks, storage and integration contracts.

What's new in 1.0.10

Release 1.0.10 follows 1.0.7 and adds four things. **A gift through a coupon code:** a promotion can list WooCommerce coupons and then counts only while one of them is applied (only with a coupon code). **A gift at a discount:** each gift can be free or cost a percentage or an amount less than its current price, for every kind of deal and every reward level (a gift at a discount). **Free shipping as the reward:** a promotion or a reward level can give free shipping through WooCommerce's own Free shipping method (free shipping as the reward). **Send a gift:** a new screen sends one customer a gift by hand as a WooCommerce order with nothing to pay, with your message (send a gift). Two corrections: while the license pauses promotions, the cart and checkout no longer show a Free Gifts box saying "Gift options could not be checked", and a gift removed by such a pause leaves no error behind. Some labels changed: **Number of gifts**, **Gift product** and, in the reports, **Gift items**. Everything else in this section came with earlier releases.

Release 1.0.5 gives the plugin its new name, AffStackDev Free Gifts & BOGO (it was called Freebie Promotions). The name changes on its screens, in the plugin list and in this guide; shoppers see plain words such as **Free gift offer** and **Free gift progress**. Promotions, settings, licenses and web addresses stay as they are.

Release 1.0.3 follows the launch release 1.0.1.3a. Two corrections for shoppers. Under **Free gift progress** on the cart page, a promotion that asks for a spending amount showed the currency symbol as a character code instead of the symbol itself; the amount now reads correctly ("Add \$7.00 more to unlock ..."). A customer with an AffStackDev B2B business account or application does not receive consumer gifts; their cart and checkout showed "Gift options could not be checked. Refresh your cart to try again." That customer now sees no gift box at all. A gift card bought in the cart (AffStackDev Store Credit) no longer counts toward a promotion unless you tick **Gift cards** under **Settings**, and a gift card cannot be a free gift: the editor refuses it, and a promotion that already names one treats it as unavailable. With WordPress debugging on, Free Gifts & BOGO's own pages no longer make WordPress log a notice when the Suite shows them. The guide built into the plugin now describes the current release, and version numbers have three parts from this release on. Everything else in this section came with 1.0.1.3a.

Release 1.0.1.3a was the AffStackDev launch release of Free Gifts & BOGO; WordPress showed it as version 1.0.1.3.1. It contains every change since 1.0.1.2.

Licensed AffStackDev products are never gifts. A product variation that AffStackDev License Server sells and Commerce Bridge protects, including a free license such as Theme Manager, cannot be a free gift. Free Gifts & BOGO refuses to save or enable a promotion whose automatic gift, named gift choices or same-product deal uses one, and the editor says why. A promotion that already names one treats that gift as unavailable: it is dropped without a substitute and the paid purchase checks out normally. Licensed products never appear among a shopper's gift choices; buying one can still count toward a promotion that rewards a different product (see Set what the customer receives).

Store credit and gift cards are payment. Paying with AffStackDev store credit or a gift card no longer lowers a promotion's eligible spending, before or after coupons, and does not count as a coupon for a promotion that does not combine with coupons: a \$60 cart paid partly with \$20 of store credit keeps its spend-50 gift. Ordinary coupons still count as configured (see [Qualification, minimum spend and coupons](#)).

Clearer messages. Without an active Free Gifts & BOGO license, **Add promotion** shows the normal license notice instead of an error page. A gift that another plugin refuses no longer leaves that plugin's error notices with the shopper, and the "A free gift could not be validated" notice is informational instead of blocking checkout.

Current Commerce Core. Free Gifts & BOGO bundles Commerce Core 0.1.0.2c, the same copy as AffStackDev Suite, Store Credit and Dynamic Pricing.

Your license and updates. Each Free Gifts & BOGO license covers one live (Production) site and one test copy of it; an installation set to Staging, Development or Local uses the license's staging place. AffStackDev's current plans do not require a Suite license, and AffStackDev Suite needs no key of its own. Download the plugin from **My Account → Downloads** on affstackdev.com; after activation, WordPress offers new releases through protected updates.

Contents

- [What's new in 1.0.10](#)
- [Work with Free Gifts & BOGO through AI](#)
- [Feature index](#)
- [What Free Gifts & BOGO manages](#)
- [Install and open the plugin](#)
- [Updates and retained data](#)
- [Native records permissions and language](#)
- [Find your promotion workspace](#)
- [Who can change promotions](#)
- [Enhanced and Performance views](#)
- [Search and review saved promotions](#)
- [Create, save, copy and remove a promotion](#)
- [Choose the kind of deal](#)
- [Set which purchases count](#)
- [Set what the customer receives](#)
- [Build a reward ladder](#)
- [Exclusions, shopper conditions and limits](#)
- [Choose where the offer is explained](#)
- [Saved values and input boundaries](#)
- [Use named products, categories and groupings](#)
- [Set global defaults](#)
- [Create or select the policy page](#)

- [Create and edit basic WooCommerce coupons](#)
- [Send a customer a gift by hand](#)
- [Read the dashboard](#)
- [Compare promotion reports](#)
- [Preview a saved promotion](#)
- [Resolve validation and action messages](#)
- [How a promotion becomes a gift](#)
- [Amounts, quantities and repeated reward sets](#)
- [Which purchases count](#)
- [When a promotion is active](#)
- [Qualification, minimum spend and coupons](#)
- [Reward levels: highest reached only](#)
- [Get the same product free](#)
- [Combining promotions](#)
- [Limiting the value of each gift](#)
- [Choosing a gift pool](#)
- [Why a product can be rejected as a gift](#)
- [Catalog reads and availability changes](#)
- [Gift-line identity and retained information](#)
- [What happens when the cart changes](#)
- [Guest, account and previous-order restrictions](#)
- [Understanding a missing or removed gift](#)
- [How gifts reach the cart](#)
- [What the cart uses to qualify](#)
- [Recalculation and changed carts](#)
- [Choosing and changing a gift](#)
- [Using a classic cart](#)
- [Store API behavior for shopping integrations](#)
- [Using the Cart and Checkout Blocks picker](#)
- [Gift records on an order](#)
- [Campaign and customer allowances](#)
- [Stock and refund review](#)
- [Checkout messages and recovery](#)
- [Choose where shoppers see promotions](#)
- [Read cart progress and remaining requirements](#)
- [Preview a saved promotion before enabling it](#)
- [Understand promotion reports](#)

- [Resolve display preview and report problems](#)
- [License status and promotion activity](#)
- [When your plan requires Suite](#)
- [Open and read License & updates](#)
- [Activate or replace the Free Gifts & BOGO license](#)
- [Checks, grace periods and recovery states](#)
- [Move a site or recover an unreadable credential](#)
- [Release this site's remote activation](#)
- [Protected plugin updates](#)
- [Work inside Suite or use the native pages](#)
- [Retained license data and uninstall choices](#)
- [Free Gifts & BOGO's Commerce Core connection](#)
- [Read Integration status](#)
- [How current eligibility controls gifts](#)
- [Protected gifts and cart recovery](#)
- [Free gift offers in a customer-referral program](#)
- [Referral compatibility on existing orders](#)
- [Resolve integration messages](#)
- [What Commerce Core does](#)
- [Read shared records and technical details](#)
- [Download reports and individual order records](#)
- [Customer eligibility and pricing integrations](#)
- [Shared cart lines and protected merchandise](#)
- [Accepted order history and publication recovery](#)
- [Diagnostic reports and integration interfaces](#)
- [Retained commerce records and safe updates](#)
- [Choosing a supported feature provider](#)
- [Shared configuration, pauses and ownership](#)
- [Fresh evaluation and short-lived permission](#)
- [One business operation, one retained identity](#)
- [Reconciling unfinished work](#)
- [Retiring a business policy safely](#)
- [Discovering checkout credit participation](#)
- [Shared operation troubleshooting](#)
- [Use the complete installed references](#)
- [Glossary](#)

Feature index

Capability or reference	Section
Work with Free Gifts & BOGO through AI	Open reference
A gift at a discount	Open reference
Accepted order history and publication recovery	Open reference
Account changes during checkout	Open reference
Activate or replace the Free Gifts & BOGO license	Open reference
Amounts, quantities and repeated reward sets	Open reference
API selection failures	Open reference
Automatic retries and safe visits	Open reference
Build a reward ladder	Open reference
Campaign and customer allowances	Open reference
Catalog reads and availability changes	Open reference
Checkout messages and recovery	Open reference
Checks, grace periods and recovery states	Open reference
Choose the kind of deal	Open reference
Choose where shoppers see promotions	Open reference
Choose where the offer is explained	Open reference
Choosing a gift pool	Open reference
Choosing a supported feature provider	Open reference
Choosing and changing a gift	Open reference
Combining promotions	Open reference
Commercial facts without price changes	Open reference
Compare promotion reports	Open reference
Confirmation and status labels	Open reference
Coupon status, Trash and store credit	Open reference
Create and edit basic WooCommerce coupons	Open reference
Create or select the policy page	Open reference
Create, save, copy and remove a promotion	Open reference
Customer eligibility and pricing integrations	Open reference
Deactivate or remove Free Gifts & BOGO	Open reference
Diagnostic reports and integration interfaces	Open reference

Capability or reference	Section
Discovering checkout credit participation	Open reference
Download either manual	Open reference
Download reports and individual order records	Open reference
Draft, publishing and committed	Open reference
Enhanced and Performance views	Open reference
Exclusions, shopper conditions and limits	Open reference
Find your promotion workspace	Open reference
Free gift offers in a customer-referral program	Open reference
Free Gifts & BOGO's Commerce Core connection	Open reference
Free shipping as the reward	Open reference
Fresh evaluation and short-lived permission	Open reference
Get the same product free	Open reference
Gift records on an order	Open reference
Gift-line identity and retained information	Open reference
Gifts that match the purchased product	Open reference
Guest, account and previous-order restrictions	Open reference
How a promotion becomes a gift	Open reference
How current eligibility controls gifts	Open reference
How gifts reach the cart	Open reference
Information shown to the shopper	Open reference
Install and open the plugin	Open reference
Interpret refunds and currencies	Open reference
Interpret reward summaries	Open reference
Language and browser assets	Open reference
Legacy B2B and Dynamic Pricing	Open reference
License status and promotion activity	Open reference
Limiting the value of each gift	Open reference
Move a site or recover an unreadable credential	Open reference
Moving between Suite screens safely	Open reference
Native records permissions and language	Open reference
One business operation, one retained identity	Open reference

Capability or reference	Section
Only with a coupon code	Open reference
Open and read License & updates	Open reference
Payment retries, cancellation and refunds	Open reference
Preview a saved promotion	Open reference
Preview a saved promotion before enabling it	Open reference
Protected gifts and cart recovery	Open reference
Protected plugin updates	Open reference
Publish a Current Promotions page	Open reference
Qualification, minimum spend and coupons	Open reference
Read cart progress and remaining requirements	Open reference
Read each report value	Open reference
Read Integration status	Open reference
Read product offers and catalog badges	Open reference
Read shared records and technical details	Open reference
Read the dashboard	Open reference
Recalculation and changed carts	Open reference
Recognize partial results and cached reports	Open reference
Reconciling unfinished work	Open reference
Recover a stopped update	Open reference
Referral compatibility on existing orders	Open reference
Release this site's remote activation	Open reference
Resolve display preview and report problems	Open reference
Resolve integration messages	Open reference
Resolve validation and action messages	Open reference
Respond to connection messages	Open reference
Retained commerce records and safe updates	Open reference
Retained license data and uninstall choices	Open reference
Retiring a business policy safely	Open reference
Reward levels: highest reached only	Open reference
Saved values and input boundaries	Open reference
Search and review saved promotions	Open reference

Capability or reference	Section
Send a customer a gift by hand	Open reference
Set global defaults	Open reference
Set what the customer receives	Open reference
Set which purchases count	Open reference
Shared cart lines and protected merchandise	Open reference
Shared configuration, pauses and ownership	Open reference
Shared operation troubleshooting	Open reference
Smaller screens and keyboard use	Open reference
Stock and refund review	Open reference
Store API behavior for shopping integrations	Open reference
Technical reference for administrators	Open reference
Understand each status	Open reference
Understand promotion reports	Open reference
Understanding a missing or removed gift	Open reference
Updates and retained data	Open reference
Use named products, categories and groupings	Open reference
Use the complete installed references	Open reference
Use the manuals with AffStackDev AI	Open reference
Using a classic cart	Open reference
Using the Cart and Checkout Blocks picker	Open reference
What Commerce Core does	Open reference
What Free Gifts & BOGO manages	Open reference
What happens when the cart changes	Open reference
What remains available during a hold	Open reference
What remains reserved	Open reference
What the cart uses to qualify	Open reference
When a promotion is active	Open reference
When B2B has been deliberately retired	Open reference
When your plan requires Suite	Open reference
Where lifecycle information appears	Open reference
Which purchases count	Open reference

Capability or reference	Section
Who can change promotions	Open reference
Why a product can be rejected as a gift	Open reference
Work inside Suite or use the native pages	Open reference

What Free Gifts & BOGO manages

AffStackDev Free Gifts & BOGO lets your customers earn free products or variations through qualifying purchases. You can award a specified gift automatically or let the customer choose from eligible products. Spend and quantity thresholds, reward levels, product restrictions, customer conditions, schedules, combining rules and campaign allowances determine which offers are available.

The plugin gives you a promotion workspace, rule builder, saved-rule preview, settings, coupon center, reports and customer-facing promotion displays. Its references describe every supported control and operating path, including the classic Cart gift form, Cart/Checkout Blocks picker, order attribution, license recovery and compatible product integrations.

Paid purchases remain ordinary WooCommerce items. Free Gifts & BOGO adds a separate identified gift line, so a customer can buy a product and also receive a free copy. WooCommerce continues to manage stock, payments, taxes, refunds and ordinary coupons. Free Gifts & BOGO records which promotion awarded a gift and maintains any campaign allowance claims. A gift does not become a Store Credit balance or referral payment.

AffStackDev Suite can place Free Gifts & BOGO inside its shared workspace. Free Gifts & BOGO works without Suite; AffStackDev's current plans do not require a Suite license, and Suite needs no key of its own. B2B customer eligibility, protected products, current Free Gifts & BOGO permission and available campaign allowances are checked separately.

Install and open the plugin

Use WordPress 6.7 or newer, WooCommerce 10.0 or newer, and PHP 7.4 or newer. The plugin needs WooCommerce installed and active. Its bundled shared Commerce Core has a separate requirement for 64-bit PHP 8.1 or newer, explained in the integration reference. Your host also supplies the encryption and signature-verification extensions described in the license chapter.

1. Download the Free Gifts & BOGO ZIP from **My Account → Downloads** on [affstackdev.com](#). In WordPress, open **Plugins → Add New → Upload Plugin** and upload that ZIP.
2. Activate **AffStackDev Free Gifts & BOGO**. When updating an existing installation, replace the plugin in place to preserve its records.
3. Open **Settings → Free Gifts & BOGO License** and activate the product's key for the correct site environment.
4. Open the Free Gifts & BOGO workspace. A compatible active Suite can place it under **AffStackDev → Free Gifts & BOGO**; otherwise use Free Gifts & BOGO's native menu.
5. Open **Settings** to configure shared defaults, then use **Promotions** to create or manage your offers. The subsequent chapters explain every setting and operation.

If activation reports **Plugin activation failed**, read the accompanying dependency message. Update the identified WordPress, PHP or WooCommerce component, or activate WooCommerce, before activating Free Gifts & BOGO again. When an already active plugin encounters missing requirements, its ordinary promotion application does not start. Native licensing recovery is loaded separately. Having plugin files present does not by itself mean the store can award gifts.

The normal merchant controls require WordPress's `manage_woocommerce` permission. A user who can manage plugins is not automatically entitled to manage promotions if that WooCommerce capability is missing. Product licensing actions use the separate administrative permissions described in their chapter.

Updates and retained data

The current product release is **1.0.3**. Its data version is 3 and rule schema is 7. The bundled Commerce Core remains **0.1.0.2c**, with native version **0.1.0.2.3**. These numbers identify different components; updating the product does not rename Core or reset your campaigns.

Activation and normal startup install missing owned campaign storage when a data upgrade is due. Existing rule formats are normalized as they are read, rather than having every campaign or historical order rewritten. If campaign storage cannot be installed, limited promotions remain subject to the storage checks in the campaign chapter. Ask the host to correct the database requirement; changing a version option is not a storage repair.

The plugin uses WooCommerce order interfaces and declares compatibility with its high-performance order storage. It provides its own Blocks gift picker and Store API integration as described later. Choose the instructions for your actual cart layout: classic Cart and Cart/Checkout Blocks have different selection interfaces.

The product does not automatically set up every site in a multisite network. Manage activation, platform requirements, configuration and storage for each site. The shared runtime also treats records as belonging to their original site.

Deactivate or remove Free Gifts & BOGO

Deactivating Free Gifts & BOGO in WordPress preserves its saved promotions, settings, credentials, campaign history and order records. Its scheduled license validation is cleared. Deactivation does not release the site's remote license activation; use the license screen's supported deactivation action when you want to free that activation.

Deleting the plugin retains data by default. If you explicitly enable the deletion checkbox under **Uninstall behavior** in its settings, uninstall removes plugin-owned promotion records, settings, licensing state, update state and the specified report caches. The cleanup handles at most 10,000 returned promotion records in one run. WordPress filters and deletion hooks still apply, so this setting is not an unlimited database wipe.

Even with that option enabled, campaign allowance claims and their schema marker remain for order retries, delayed payments and reinstall safety. WooCommerce orders, their saved gift attribution, coupons, products, customers and other plugins' records remain. Deleting configuration does not reset recorded campaign usage, reverse refunds or remove a remote license activation. Review the option before deleting the plugin if you intend to reinstall and keep using its campaigns.

Native records permissions and language

Promotions are private WordPress records. They are not public pages, searchable storefront posts, navigation-menu items or public REST content. The product's promotion workspace provides the normal management screens. A campaign supports a title, editor content, excerpt and revisions; its versioned rule configuration also participates in revisions.

Every native promotion capability maps to WooCommerce management permission, including creation, editing another user's campaign, publication, private-record access and deletion. A campaign's creator does not gain special access from authorship alone. Deleting a creator does not automatically delete their campaigns.

The post type permits WordPress's own authorized export tools. This is distinct from the Free Gifts & BOGO report screens and does not add a Free Gifts & BOGO report-export button. Follow the merchant reference for the actions actually present in each screen.

Language and browser assets

Free Gifts & BOGO loads its text domain through WordPress's normal translation system and plugin-relative language directory. Use available WordPress or supplied translation resources for freebie-promotions-for-woocommerce. Core uses its own translation domain. The plugin has no separate language selector or automatic translation service.

The installed JavaScript and style files are ready for WordPress to load. Merchants do not need to compile assets or run a package manager. Performance and Enhanced modes use the same saved product records; their navigation and browser behavior are explained in the merchant chapter.

Technical reference for administrators

The technical startup reference explains exact platform issues and initialization order. The storage reference supplies post-type and capability mappings. The uninstall reference lists every cleanup target and retention boundary for administrators managing backups, migrations or removal.

Find your promotion workspace

Open **Free Gifts & BOGO** in WordPress administration. If your store uses a compatible AffStackDev Suite, open **AffStackDev → Free Gifts & BOGO** instead. The workspace contains **Dashboard, Promotions, Reports, Coupons, Send a gift, Preview, Settings** and **License**. The same saved promotions are used in both locations.

Start in Settings to choose the shared defaults, then use Promotions → Add promotion to create a draft. Configure what customers must buy and what they receive, save, and use Preview saved version to check a sample. Choose Published and enable the rule when it should run. The selected schedule still applies.

Destination	What you can do
Dashboard	See current promotion states, recent order totals, leading promotions and coupon count; add a promotion.
Promotions	Search/filter saved promotions, edit their full rules, enable or disable, duplicate, move to Trash, restore or permanently delete.
Reports	Compare recorded promotion orders and gift products for a selected period; open the underlying order or product.

Destination	What you can do
Coupons	Create and edit basic WooCommerce coupons, manage status or open their advanced owner settings.
Send a gift	Send a customer a gift by hand: a free WooCommerce order with your message. See send a gift .
Preview	Calculate one saved promotion against one sample product line and a selected time.
Settings	Choose global exclusions, default behavior, wording, presentation mode, policy page, refund-note and uninstall preferences.
License	Manage product access and recovery; see license administration .

Who can change promotions

The workspace requires WooCommerce merchant-management permission. Creating or changing promotions, coupons through this center, or the policy page also requires current product permission. Existing records, reports and preview remain available when new promotion activity is held. A displayed button does not override a license or account restriction.

Creating a policy page additionally requires permission to edit WordPress pages. Coupon creation requires coupon-editing permission, and editing/deleting an existing coupon requires permission for that record. If an action says it is not permitted or has expired, reopen its current screen and use a fresh control after confirming your access.

During a licensing hold, the ordinary native PHP Settings form can still change Admin display mode, Refund review warning and Uninstall behavior. Other promotion settings stay retained. An asynchronous Suite settings save is blocked in that state; open the native PHP settings page for those three preferences. The [Documentation card](#) remains available.

If the plugin does not start, administrators able to activate plugins see the missing requirement: WordPress 6.7 or newer, PHP 7.4 or newer, and installed/active WooCommerce 10.0 or newer. Resolve the named requirement before using the workspace.

Enhanced and Performance views

Settings → Admin display mode offers Enhanced AffStackDev workspace and Performance / PHP pages. Both keep the same forms and saved promotion behavior. Enhanced adds conditional fields, live summaries, instant reward-level editing and in-page Suite navigation. Performance uses ordinary page loads and shows the server-rendered controls. WooCommerce's product search still needs browser JavaScript; existing selections remain visible without it.

The native promotion header can offer **Use performance view** or **Use enhanced view** for the current request. This does not change the saved setting. If the product or Suite parent has enhancement turned off, the link opens Presentation settings instead. A request cannot turn on a layer the saved preferences have disabled.

Search product names or SKUs using at least two characters in the enhanced picker. Select exact variation options for automatic gifts. Categories, tags, types and groupings use named selections. The summary changes

as you edit, but you must save to apply changes. Fields hidden by your deal choice can keep their previous values; review limits and exclusions when changing a deal's kind.

Moving between Suite screens safely

The workspace warns before discarding unsaved form changes. While an in-page action is pending, controls are temporarily locked and navigation is held until the response returns. Preview is also treated as a guarded request in this interface even though it does not change a cart. Back and Forward restore the relevant screen and filters. Links to native WordPress or WooCommerce tools can prompt to open a new tab.

If a request reports that the screen could not update, keep your entered information and inspect the saved record in another tab before repeating the action. A server write can complete even if the response fails to reach the browser. Promotion saves detect a changed saved version; coupon saves do not have that same protection. Native PHP errors offer their normal page/back route.

Enhanced mode asks before coupon Trash and permanent deletion. Performance mode does not provide that JavaScript confirmation; the authorized coupon action link can act immediately. Promotion permanent deletion has its own separate confirmation page in either mode.

Smaller screens and keyboard use

Navigation, forms and cards rearrange for a narrow screen or Suite panel. Wide report, promotion and coupon tables scroll sideways so their columns remain available. The Promotions table's scroll region can receive keyboard focus. Buttons and links show a focus outline; status is written in text as well as color. Enhanced validation highlights the first field it can locate and opens its advanced section. If an error link does not move to a customer-targeting or campaign-limit control, open **Exclusions and limits** and use the named field. Reduced-motion preferences are respected by the enhanced transitions and scrolling.

Search and review saved promotions

Open Promotions, choose **Show**, enter **Search promotions** if needed, and select **Apply**. Views are All promotions, Drafts, Active, Upcoming, Expired, Disabled and Trash. The list shows 20 records per page, most recently modified first. Search accepts up to 100 bytes of text. Use Previous and Next; the display shows the current page, page count and matching total.

Each row identifies the promotion, status/priority, what the customer must do, what they receive and the schedule. Edit opens the full embedded form; Preview opens the saved version. Duplicate makes a separate disabled draft. Enable/Disable changes the enabled flag; Move to Trash removes the rule from normal use. Trash offers Restore as draft and Delete permanently.

The native WordPress promotion list remains available through WordPress tools. It adds Active, Upcoming, Expired and Disabled state filters, Preview and Global settings links, plus bulk Enable promotions and Disable promotions. Quick Edit is not available. Bulk enabling skips drafts, invalid configurations, missing/trashed records and records you cannot edit; it does not publish drafts. Read the updated and skipped counts afterward.

Understand each status

Status	Meaning and next step
Not published	The post is not Published. A completed draft does not run. Choose Published when ready.

Status	Meaning and next step
Disabled	The enabled flag is off, including a rule saved disabled after validation. Review and enable it when appropriate.
Needs attention	A published enabled configuration fails the current validation. Open and correct it. Some malformed saved data normalizes to Disabled instead.
Upcoming	Published, enabled and valid, but its start time is later.
Active	Published, enabled, valid and within its schedule. Shopper access, license, stock and allowance checks still apply.
Expired	Its end boundary has been reached. Adjust the schedule if the offer should return.
In Trash	The rule is in WordPress Trash. Restore creates a disabled draft for review.
Removed in reports	Historical order evidence remains, but the promotion record no longer exists.

Active counts are current rule states, not a promise that every shopper can receive a gift. State counts and live rule discovery are bounded to 5,000 records by default; an integration can adjust that bound within 200–10,000. Very large stores should account for those bounds when reviewing completeness.

Create, save, copy and remove a promotion

1. Select **Add promotion**. This reserves an unpublished draft and opens its editor.
2. Enter **Promotion name**, optional **Short description** and **Full description**. The descriptions support basic HTML. Internal notes have their own field and are for managers.
3. Choose the deal and complete Customer must, Customer gets, and Exclusions and limits.
4. Set the schedule, display choices and Publication. Leave the rule disabled or in Draft while preparing it.
5. Select **Save promotion**, correct any validation messages, then use **Preview saved version**.
6. When ready, choose Published, turn on Enable this rule and save. A future start time still delays the offer.

A validation failure can save your content and rule as disabled. Reopen the named controls, correct the values and save again; saving the same invalid form will not activate it. If a record changed after you opened it, the embedded editor refuses your stale submission. Open the current promotion in another tab, compare your edits, then work from the fresh version. It does not merge changes automatically.

Duplicate copies the descriptions and rule into a new promotion named with “(Copy)”, always a disabled draft. Review products, dates, targeting and limits before publishing the copy. The new promotion has its own identity; source campaign usage, order history and revision history are not copied.

Disable retains the rule. **Move to Trash** uses WordPress Trash and is unavailable in the embedded promotion screen if Trash is disabled. To bring it back, select Trash → Restore as draft, then review and republish/enable. To remove it permanently, open Delete permanently, read the identified promotion, check **Yes, permanently delete this promotion**, and confirm. This removes its rule and revisions; existing order records stay. Cancel returns to Trash. If a restore or deletion reports a storage error, inspect the current record before trying again.

WordPress tools → Open WordPress editor provides native revision history, media and other WordPress features for the same record. The native editor uses title for the name, Excerpt for the short description and the main editor for full customer copy.

Choose the kind of deal

The editor presents seven deal choices. Start with the one closest to your offer, then review the fields that appear. Several choices share the same underlying rule, so reopening a saved customer-choice offer may show the broader catalog-choice label while preserving its quantity or spend behavior.

What kind of deal?	Setup
Customer picks a free gift from the catalog	Choose spending an amount or buying a number of items, then allow all eligible gifts or a selected pool.
Spend an amount, get a reward	Enter a spend threshold and choose the reward method: a product added automatically, the customer's choice, the same product, or <u>free shipping</u> .
Buy products, get a specific product free	Set the qualifying quantity and exact automatic gift product/variation.
Buy products, choose free products	Set the qualifying quantity and the shopper's gift choices.
Buy products, get the same product free	Set qualifying purchases and quantity; each exact product/variation earns its matching free item.
Mix and match products, choose free products	Combine purchases from the selected scope and let shoppers choose gifts from that same scope.
Spend or buy more, get different rewards	Build at least two levels, each with its own threshold and reward; only the highest reached level applies.

A new rule starts disabled, with evaluation order 10, no dates, spend qualification, all-products scope, one automatic reward, one reward set per order, empty selections and no campaign caps. Zero thresholds and an unselected automatic product must be completed before enabling it.

Evaluation order accepts -9,999 to 9,999; smaller numbers win when promotions compete, with an older promotion winning a tie. **Starts** and **Ends** use the displayed site timezone. Blank means no boundary, and End must be later than Start. Saved instants remain stable if the site timezone later changes. **Internal notes** are separate from customer descriptions. The Promotion summary is a live explanation in Enhanced mode, not a save or stock guarantee.

Set which purchases count

Control	How to use it
Customer earns the reward by	Select Spending an amount or Buying a number of items when that choice is offered. Other deal templates set their appropriate basis.
Amount to spend	Enter a positive amount for a single spend offer. Qualifying merchandise counts; tax, shipping, fees and free gift lines do not.
Number of items	Enter at least one for a single quantity offer.
Products that count	All products except exclusions, or Only selected products or groups. Selected requires at least one selection.
Specific products	Search names or SKUs; choose an exact variation where needed.

Control	How to use it
Product categories	Select qualifying categories.
Product tags	Select qualifying tags.
Attributes and groupings	Select named colors, sizes, brands or other registered product groupings.
Product types	Select the WooCommerce product types that count.
Purchases that do not count	Exclude products from qualification only. A parent excludes all its variations; an exact variation excludes only itself.
Purchase categories that do not count	Exclude a category and its child categories from qualification only.
Also require a minimum spend	Turn on an additional amount gate, usually alongside a quantity offer.
Additional minimum amount	Enter greater than zero when that switch is on.
Maximum reward sets per order	Default 1 awards the deal once. A larger integer caps repetitions; 0 permits repetition each time the qualifying threshold is reached. This is separate from gifts per set and campaign-wide limits.

Selected products and groups combine: a purchase may qualify through any selected grouping. Exclusions still win. Purchase-only exclusions can still be offered as gifts unless another exclusion blocks them. A tiered offer uses its highest reached level once rather than repeating the single-level reward-set setting. See [rule evaluation](#) for examples and [catalog matching](#) for category and variation details.

Set what the customer receives

Control	How to use it
How the reward is chosen	Add a specific product automatically, let the customer choose, give the same product they bought, or give free shipping (free shipping as the reward). Deal templates select the applicable method. Same-product offers require quantity qualification.
Number of gifts	At least one; this is the number of slots per earned reward set.
Gift product	Choose the exact automatic product or variation. A variable parent is not a concrete gift selection.
Customer can choose from	All eligible products, Only selected products or groups, or the same products/groups that count toward the deal. Mix-and-match uses the latter automatically.
Allowed products	Select named products/variations for a curated gift pool.
Allowed categories	Select categories in the gift pool.
Allowed tags	Select gift tags.
Allowed attributes or groupings	Select gift colors, sizes, brands or other available groupings.
Allowed product types	Include types in a curated union, or restrict types when offered with the all-eligible pool.

Control	How to use it
Limit reward value	Turn on to reject gift products above a chosen value.
Maximum reward value	Enter a positive per-product maximum.
Price used for the limit	Use the global default, current active price or regular price.
Price of each gift	Free (the start), A percentage off its price, or An amount off its price. See a gift at a discount .
Discount	For a percentage, a number above 0 and below 100; for an amount, the amount off in your store currency. Shown for the two discount choices.
Products never offered as gifts	Exclude a parent/all its variations or an exact variation from rewards only. Paid purchases can still qualify.
Categories never offered as gifts	Exclude that category and descendants from rewards, including automatic gifts and every level.

Curated products and groupings combine by OR. A selected category or product is not a promise of available stock; gifts must satisfy the current candidate rules. Global/shared exclusions still apply to both purchases and rewards. A value limit is per candidate, not an aggregate campaign budget. With no value limit enabled, the stored Regular choice does not replace the current-price reference used by the current gift validator.

Licensed AffStackDev software cannot be a gift. A product variation that License Server sells and Commerce Bridge protects, including a free license, is refused when you save or enable a promotion whose automatic gift, named gift choices or same-product deal uses it; the editor explains which choice to change. Licensed products are left out of every gift pool, and a promotion that already names one drops that gift without a substitute while the paid purchase checks out. A licensed product can still count toward the purchases that qualify for a different gift.

A gift at a discount

Under **Gift price**, **Price of each gift** decides what the shopper pays for each gift: *Free* (the start), *A percentage off its price* or *An amount off its price*, with the **Discount** below it. The same choice is on every reward level of a ladder, and it works for every kind of deal: a product added automatically, the customer's choice, and the same product (buy one, get one at 50% off).

- The discount comes off the gift product's current price, its sale price while it is on sale. 50 means 50% off: a \$10 gift costs \$5.00. An amount at or above the price makes the gift free.
- A percentage discount is rounded to the cent the way WooCommerce rounds a percentage coupon: 50% of \$9.99 is a \$5.00 discount, so the gift costs \$4.99.
- If the product's price changes while the gift is in a cart, the gift's price follows at the next cart update.
- WooCommerce coupons apply to a gift at a discount like to any other product (a 10% coupon makes a \$5.00 gift \$4.50); **Coupon coexistence** still decides whether coupons may be used with the promotion at all. Store credit and gift cards pay for it like for any product, and the store's tax settings apply to it.
- The gift still never counts toward a promotion, and Dynamic Pricing never changes its price.
- **Limit reward value** still compares the product's own value, not its reduced price.

- Shoppers read **Gift at 50% off** (or **Gift at \$5.00 off**) on the gift line and in the order; product pages and the promotions page say "Get 1 × Socks at 50% off"; a choice shows each gift's price, such as "\$4.99 (50% off)". Cart messages say "gift" instead of "free gift" for these gifts.
- The order records the gift's price with the gift; campaign limits count it like a free gift, and the reports count it under **Gift items issued**.

Customer-choice gifts are selected in the cart/checkout picker. Buying an item normally from the catalog does not turn it into a free item. Same-product offers keep exact variations separate, whereas mix-and-match combines matching purchases. See [same-product gifts](#), [candidate availability](#) and [gift value limits](#).

Free shipping as the reward

Choose **Give free shipping** under **How the reward is chosen** (on a level of a ladder: **Free shipping**) to give the customer free shipping instead of a gift. Free Gifts & BOGO sets no shipping price itself: it works through WooCommerce's own **Free shipping** method.

1. Under WooCommerce > Settings > Shipping, add the **Free shipping** method to each shipping zone where the promotion should give free shipping, and set its **Free shipping requires** to *A valid free shipping coupon, A minimum order amount OR coupon* or *A minimum order amount AND coupon*.
 2. Build the promotion with **Give free shipping**. The purchases that count, shopper conditions, schedule, coupon rules and limits work as for a gift; there is no gift product, number of gifts or gift price to set.
- While the cart reaches the promotion, it counts as a valid free shipping coupon for those Free shipping methods, so WooCommerce offers Free shipping at checkout next to the store's other methods, and the customer chooses it like any shipping method. With *A minimum order amount AND coupon*, WooCommerce's minimum still has to be met.
 - A Free shipping method set to *A minimum order amount* or *No requirement* works as before, and so do all other shipping methods and their prices. A zone without a Free shipping method gives no free shipping.
 - When the cart no longer reaches the promotion, Free shipping is no longer offered and WooCommerce moves the cart to another method.
 - A promotion gives free shipping once per order, however often the cart qualifies.
 - Under the reward choice the editor says how it works, and warns when no shipping zone has an enabled Free shipping method that accepts a coupon.
 - The order records the promotions that gave its free shipping, only when WooCommerce's own free shipping coupon or minimum amount would not have given it. The order screen's **Gift promotions** section and the customer's order details and emails list it as "Free shipping over \$50: Free shipping". Campaign limits count one gift unit for each such order, and the reports count the order for the promotion, not as a gift item.
 - Business (B2B) customers get no consumer promotion, so no free shipping from one either.

Build a reward ladder

Choose **Spend or buy more, get different rewards**, select the spend or quantity basis, and complete at least two reward levels. Each level has Customer spends or Customer buys this many, Number of gifts, and Customer gets. Choose an automatic concrete product, a customer-choice pool or **Free shipping** ([free shipping as the reward](#)) at each level. A choice pool can use all eligible products or the level's own Specific products, Categories, Tags, Attributes and groupings, and Product types.

Each level has its own **Price of each gift** and **Discount** (a gift at a discount): for example 50% off from \$100 and free from \$200. Expand **Optional maximum gift value** to enable a value cap for that level, enter Maximum value and choose Price used. Shared rule exclusions, freebie-only exclusions, customer conditions and schedule apply to every level. Same-product gifts and the “same qualifying scope” pool are not separate tier choices.

Use **Add another reward level** or **Remove level** to change the ladder; at most 20 submitted levels are retained. Thresholds must be positive and distinct. Cards with both thresholds blank/zero are ignored. Saved levels sort by their active threshold, so the displayed order can change on saving. Only the highest reached level supplies the reward; lower levels are replaced rather than accumulated.

Enhanced Add/Remove changes the unsaved form. Save afterward. In Performance or without the product's enhancement script, these buttons submit and save the entered form; Add supplies a next threshold one above the current maximum's whole-number part. The new level can leave the promotion disabled until its gift is configured. Always review the saved ladder and preview its thresholds.

Exclusions, shopper conditions and limits

Under **Exclusions and limits**, Exclude products, Exclude categories, Exclude tags, Exclude product types, and Exclude attributes or groupings block both qualification and gifts. Use the purchase-only and freebie-only lists when only one side should change. Those two category-exclusion controls include descendants; the older shared/global category controls keep their direct-category behavior.

Customer targeting control	Options and consequences
Shopper access	All eligible shoppers (default), Guest checkout only, or Signed-in customers only.
Customer roles	Leave empty for any role; otherwise select one or more named roles using Ctrl/Command where needed. Any selected role can match, but sign-in is required. Up to 50 submitted role entries are accepted.
Account purchase history	Any (default), No previous paid orders, or At least one previous paid order. History conditions require sign-in.

All selected conditions must match, and B2B/Core and license eligibility still apply. Guest-only cannot require account roles, paid-order history or a per-account limit. Paid history belongs to the signed-in account and uses recorded payment dates; later-refunded or cancelled paid orders still count. Guest orders are not matched by email. **No previous paid orders** automatically uses one campaign gift order per account. If you later remove that history condition, the saved per-account cap stays at 1 until you change it. This is checkout eligibility, not a promise that its payment is the first to settle. See customer targeting.

Campaign limit	Default and meaning
Orders across this campaign	0 unlimited; otherwise each submitted order with at least one gift uses one allowance.
Orders per signed-in customer	0 unlimited; a positive value requires sign-in and counts that account. No previous paid orders uses 1.
Gift units across this campaign	0 unlimited; otherwise limits gift units for this rule. Physical stock stays with WooCommerce.

Each limit accepts whole numbers 0–1,000,000. Usage starts with orders allocated while limits are enabled; earlier orders are not backfilled. Editing a cap does not reset usage. Pending orders reserve allowance; unpaid failed/cancelled orders release it; paid awards and refunds retain usage. A cart alone makes no reservation. The editor shows allocated orders/gift units and pending orders, or reports that allowance storage is unavailable. When unavailable, limited campaigns remain held; restore storage rather than replacing the cap with unlimited merely to clear the message.

Expand **Advanced behavior and limits** for Sale products, Virtual products and Downloadable products (each Use global default/Allow/Exclude); Coupon coexistence (global/allow/do not combine); Coupon spend basis (global/before/after discounts); and Combine with other gift promotions? (store default/yes/no). If you choose an exclusive promotion, Competition group can be blank to compete globally or the same short name as related competitors. Evaluation order controls priority. These settings are separate from a Woo coupon's "Do not combine with other coupons" flag.

Only with a coupon code

Only with these coupon codes takes existing WooCommerce coupon codes, separated by commas (empty by default). With codes listed, the promotion counts only while one of them is applied to the cart, and everything else of the promotion still has to match: a spend-10 promotion with the code *summer* needs both \$10 of qualifying products and *summer*. WooCommerce decides whether a code may be used at all (expiry date, usage limits, allowed email addresses, minimum spend) and what it discounts; a coupon may discount nothing (a fixed cart discount of 0), so the code only unlocks the gift. The codes are saved as their coupons, so renaming a coupon keeps it in the promotion.

A code that is not a published WooCommerce coupon is not kept and the promotion is saved disabled; store credit and gift card codes are payment and cannot be used. The promotion's own codes never count as "a coupon" for its **Coupon coexistence** choice; any other coupon still does. Shoppers never see the codes from Free Gifts & BOGO: product pages and the promotions page say "With a coupon code only.", and the cart shows no progress for the promotion until one of its codes is applied. Removing the code from the cart removes the gift. See [coupons and spend basis](#).

Choose where the offer is explained

The Storefront display switches are Product pages (on by default), Shop/catalog cards (off), Cart progress messaging (on) and Current Promotions policy section (on). Select the places where customers should see the offer's explanation, then save. These switches use the same rule and do not create independent deals. The native WordPress box is labeled Planned Storefront Display even though these are the current presentation controls.

Use Promotion name for the title, Short description/Excerpt for a brief explanation, and Full description/main editor for the complete terms. Internal notes do not supply customer copy. The global Customer-facing label defaults to Free Gifts. See [storefront presentation](#) and [policy presentation](#) for what qualifies to appear.

Saved values and input boundaries

The editor stores one versioned configuration with the WordPress promotion, alongside its native descriptions and revisions. Dates are saved as UTC instants. Money fields use WooCommerce decimal formatting; negative or malformed rule amounts normalize to zero, which cannot enable a positive-threshold/value-cap requirement. Enter ordinary nonnegative amounts and whole-number limits using the named controls.

Each product/category/type/grouping selection has a 5,000-entry storage bound. Very long pasted integration lists are also bounded; the normal named selectors avoid needing to type IDs. Invalid roles or malformed campaign limits disable the rule rather than silently removing the restriction. Drafts can retain incomplete thresholds/rewards for later completion. If an enabled rule has an error, it is saved disabled with an explanation; complete the named field and enable it again.

The technical schema reference gives exact programmatic input types, limits and legacy normalization. A successful save records configuration; live product eligibility, stock, customer status and campaign allowance are checked separately.

Use named products, categories and groupings

Search products by name or SKU and read variation options before selecting. For an automatic gift choose a specific variation rather than its variable parent. Category labels show their parent path; attributes and other product groupings appear under their named taxonomy. Product-type choices follow the types WooCommerce currently provides.

Saved unavailable products, groupings or types remain visible so you can review them. A category displayed as Unavailable saved selection may also be outside the selector's bounded first list, not necessarily deleted. The default list shows up to 1,000 terms per taxonomy; an integration can adjust this within 100–5,000. Rechoose a current named entry when changing an unavailable selection. Changes to the Woo catalog affect future matching; the promotion does not maintain its own copied product catalog.

Set global defaults

Open **Settings**, review the following sections and select **Save Changes**. Rules using a global/default choice read these values; an explicit rule choice can override its corresponding default. Global exclusion lists remain shared across promotions.

Control	Initial setting and use
Admin display mode	Enhanced AffStackDev workspace. Choose Performance / PHP pages for ordinary page loads with fewer product scripts.
Excluded products	Empty. Search products/variations that should neither qualify nor be offered as gifts.
Excluded categories	Empty. Select shared category exclusions; these use direct-category matching.
Excluded tags	Empty. Select tags excluded globally.
Excluded product types	Empty. Select types the promotions should ignore.
Excluded attributes or groupings	Empty. Select colors, sizes, brands or other named groups excluded globally.
Sale products	Allowed by default. Uncheck to exclude unless a rule explicitly allows them.
Virtual products	Allowed by default.
Downloadable products	Allowed by default.
Gift cards	Off. A gift card bought in the cart (AffStackDev Store Credit) counts toward no promotion. Tick it to count gift card purchases like other products. A gift card is never given as a free gift.

Control	Initial setting and use
Coupon coexistence	Allowed by default. Individual promotions and checkout policy still apply.
Reward value basis	Current active price. Choose Regular price if inherited value caps should use it.
Coupon spend basis	After coupon discounts. Choose Before if qualifying spend should be measured before coupons.
Default promotion combining	Combine promotions by default. Choose Do not combine to make inherited rules compete.
Customer-facing label	Free Gifts. Enter your preferred short wording; blank returns to Free Gifts.
Policy page	None. Select an ordinary WordPress page, then save. A draft can be selected but must be published before its customer links become available.
Refund review warning	On. Adds a private order note when a refund may affect qualification; it does not remove gifts or change refunds automatically.
Uninstall behavior	Off: retain promotion rules and settings. Turn on only when you intend those records to be deleted during uninstall. Woo products, coupons, customers, orders and campaign allowance history remain. Deactivation does not delete configuration.

The **Integration status** panel separately reports B2B / Commerce Core compatibility, current Promotion permission and Campaign allowance storage. A recognized integration does not override customer eligibility. If allowance storage is unavailable, limited campaigns cannot award gifts; ask the host to check the plugin's InnoDB table and database permissions. These local status details remain readable during a license hold and without enhanced JavaScript.

The [Documentation card](#) supplies the current manuals outside the settings form. Saving preferences is not needed to download them.

Create or select the policy page

To use an existing page, choose it in **Policy page** and Save Changes. To start a new one, use the separate **Create policy page** tool. Save any other settings first: this action reads saved settings and does not save the unsaved main form.

Creation makes a draft named **Current Free Gifts & BOGO Promotions**, with introductory wording and the live Current Promotions block. If a valid page is already selected, the tool reuses it. Choose **Edit policy page**, review the text, and publish when ready. A published page also has **View policy page**; a draft shows a publication reminder. You need permission to edit WordPress pages as well as promotion-management permission.

The page stays an ordinary WordPress page. The promotion list updates from current rules; use the rule's Current Promotions policy section switch to control its presentation. If creation fails, select an existing page or retry after resolving the page/storage issue. If a selected page was removed or moved to Trash, saving clears that selection and asks you to choose another. See [policy presentation](#).

Create and edit basic WooCommerce coupons

Open Coupons to see WooCommerce's existing coupon records. Use All, Enabled, Disabled or Trash, search a coupon code, and move through 20 results per page. Results show Code, Discount/type, Status, Uses, Expires and Actions. Enabled means Published; a coupon can still be expired, exhausted or restricted at checkout. Search starts an All view; reselect a status filter if needed.

Select **New basic coupon**, or **Basic edit** on an existing coupon. Complete these controls and choose Create coupon or Save basic changes:

Field	What to enter
Coupon code	A unique code, or leave blank to generate one. A supplied normalized code is limited to 200 bytes. Clearing an existing code generates a replacement code too.
Discount type	Percentage discount, Fixed cart discount or Fixed product discount. New coupons start with Fixed cart discount.
Amount	A nonnegative amount; percentage cannot exceed 100. Zero is allowed.
Internal description	Manager-only notes, limited to 2,000 bytes.
Expires	Optional date; expiry is set to the end of that date. Blank means no expiry.
Minimum spend	Optional WooCommerce minimum.
Maximum spend	Optional maximum; cannot be below a supplied minimum.
Total use limit	Whole number; 0 means unlimited.
Limit per customer	Whole number; 0 means unlimited. WooCommerce owns use counting.
Enabled	Checked for a new coupon. Uncheck to save it as Disabled.
Do not combine with other coupons	WooCommerce's individual-use flag. It is separate from Free Gifts & BOGO's promotion coexistence choice.
Allow free shipping when the shipping method supports it	Enables that coupon flag; a supporting shipping method is still required.

The basic form does not replace product/category restrictions, allowed email addresses, excluded sale-item settings or extension controls. Use **Advanced settings** or **Advanced edit in WooCommerce** for those. A special extension coupon displays Open advanced editor instead of the basic form. Unexposed settings on an existing basic coupon remain with WooCommerce.

If a code already exists, choose another or leave it blank for generation. If a save reports an error, reopen the coupon to check its stored values before repeating the request. Basic coupons do not have the promotion editor's stale-version check, so avoid concurrent edits to the same coupon.

Coupon status, Trash and store credit

Enable and Disable change the coupon's published/draft state. Trash sends it through WooCommerce's deletion behavior; Restore uses WordPress restoration. **Delete permanently** is offered in Trash and requires the coupon still to be there. It cannot be undone. Check the store's WordPress/WooCommerce Trash policy before using

these actions: unlike promotion management, the coupon path does not add its own “Trash disabled” safeguard.

Enhanced mode displays confirmation prompts for coupon Trash and permanent deletion. Performance mode does not run those prompts, so an authorized click can act immediately. These actions also appear for advanced extension coupon types; use the owning extension's tools when its balance or special state matters. Existing coupons remain WooCommerce records even if permission to change them through Free Gifts & BOGO is held.

The Store credit panel only recognizes registered coupon-type names suggesting a credit extension. It does not check a customer's balance or grant credit. Special credit coupons use their native editor. If no matching extension is detected, the panel provides an external WooCommerce Store Credit information link. Free Gifts & BOGO does not create a reusable credit wallet through this basic coupon form.

Send a customer a gift by hand

Open **Send a gift** to send one customer a gift without a promotion: Free Gifts & BOGO creates a free WooCommerce order for them.

1. Under **Customer**, search for the customer's account.
2. Under **Gift products**, choose up to ten products: simple products or exact variations, one of each.
3. Write a **Message** if you like (up to 1,000 characters).
4. Choose **Send gift**.

The order is an ordinary WooCommerce order with nothing to pay:

- It belongs to the customer and appears in their account. It carries their saved billing address and ships to their saved shipping address, or to their billing address when they have no separate shipping address.
- Each gift is its own line at 0, labelled **Gift sent by hand** on the order screen, where the **Gift promotions** section lists them as "Sent by hand: Candle, Card". The payment method reads "Gift from the store", and an order note names the store manager who sent it.
- The message is the order's customer note: the customer reads it in WooCommerce's order email and in their order details, where the gifts are listed under "Gift from the store".
- WooCommerce completes the order at once: it reduces stock, sets the status to Processing (Completed when nothing in it needs processing, such as a virtual downloadable product) and sends its usual order emails, the store's new-order email and the customer's order email.

The screen refuses, without creating an order: licensed AffStackDev software and gift cards, which are never gifts; variable, grouped or external products (choose a simple product or one exact variation); unpublished and out-of-stock products; more than ten products; and a customer without a saved address when a gift needs shipping. Sending the same form twice, with a double click for example, creates no second order: the screen names the first one. **Gifts sent by hand** beside the form lists the ten most recent gift orders with their customer, gifts and status. Sending needs your store-management permission, WooCommerce's permission to edit orders and an active Free Gifts & BOGO license.

Read the dashboard

Choose 7, 30, 90 or 365 days; the initial period is 30 days. Active promotions is the current state count, while Promotion orders, Gift items issued and Promotion-order value use the chosen period. Value is net of recorded

refunds. The current leader panel shows the top promotion and its orders/gifts; other panels show Needs attention, Coming up and published WooCommerce coupon count.

Use Review this promotion, Review promotions, See upcoming promotions and Manage coupons to open the relevant records. The dashboard comparison shows the top five promotions. **View full report** opens Reports at its normal default range, so choose the desired period there again. A historical promotion may have been removed; its order usage remains even if its edit link no longer opens a current rule.

The Suite dashboard widget uses a fixed 30-day view with Active, Orders, Gift items and leading promotion. If reporting is temporarily unavailable, its message does not itself mean that rule or checkout behavior has changed.

Compare promotion reports

Open Reports and choose the period. **Promotion comparison** shows promotion name, orders, gift items, average order, attributed order value, most-used reward and current status. **What customers received** lists gift products and recorded quantities, with View product links where available. **Recent promotion orders** links to orders and shows date, gift-item units and net order value. **Gifts sent by hand** counts the gift orders and gift items sent by hand in the period, apart from the promotions. A promotion that gives free shipping counts each order it gave free shipping to under its orders, with "Free shipping × 3" (for three orders) as its most-used reward; free shipping is not a gift item.

The report includes processing, completed and on-hold orders with valid Free Gifts & BOGO attribution in the rolling period. Cancelled, failed, pending and refunded-status orders are excluded. Recorded refunds reduce order value; gift quantities are recorded line quantities, not a net count of returned inventory. A partially refunded order that remains processing/completed/on-hold can still appear.

Do not add promotion-row values together as sales: stacked promotions can each receive attribution for the same order's whole net value. Current promotion status is separate from historical usage. Amounts display in the current store currency, without a separate breakdown for mixed order currencies. Use the underlying orders for accounting decisions and refunds. See report values.

The screen shows at most 250 ranked promotion rows and 250 gift-product cards. If its warning says to export or narrow the period, use the period controls or inspect the WooCommerce orders: this screen has no report-export button. A partial-report warning means the bounded scan was reached; a narrower range can help. Other service limits and caching are described under report scope and cache. Empty results have separate messages for no usage, no gifts and no recent orders. A storage error asks you to check Woo order storage and try again; it does not issue or reverse anything.

Preview a saved promotion

1. Save the promotion, then select **Preview saved version**, or open Preview and choose Promotion to test. Draft and disabled rules can be previewed.
2. Choose **Product in the sample cart**, using the exact variation where options matter.
3. Enter **Cart value before discounts** and **Cart value after discounts**. These are totals for the whole sample line, not a unit price. After cannot exceed Before.
4. Enter **Quantity in cart**, greater than zero. Use whole units for ordinary products; the form also accepts fractional quantities.

5. Choose **Date and time to test** in the store timezone. It initially shows the current minute.
6. Type the WooCommerce codes of the sample cart under **Coupon codes applied** (optional, separated by commas), check **Another coupon is applied** when the cart should also hold a coupon that is not listed, and choose **Evaluate Promotion**. A promotion with coupon codes counts only with one of its codes typed in; a code that is not a published WooCommerce coupon is reported.

Amounts initially show 0.00, quantity 1, no codes and the coupon box unchecked. The preview list displays the first 500 promotions by name. To reach another saved promotion, open it and use Preview saved version; that direct selection is retained even outside the first list. Back to promotions returns to the list.

A result can say Reward earned in this preview, No reward in this preview or Preview needs attention. It shows the saved promotion's current status, sample product, qualifying spend and quantity, and any earned reward/level. The sample time can be in the future while the saved status still says Upcoming today.

The calculation temporarily treats the saved rule as enabled and published. It still applies its schedule, product scope, exclusions, coupon rules, normal reward limits and levels. It does not change a real cart, reserve campaign allowance, reduce stock or activate the rule. It also does not select an account or establish shopper access, roles or purchase history; live cart and checkout enforce those conditions. A sample reward does not guarantee live candidate availability or checkout admission. See [preview interpretation](#).

Resolve validation and action messages

For a saved-disabled rule, read the listed field message and correct that field. Common causes are an empty name, zero qualification threshold, empty selected scope, missing automatic gift, empty curated pool, a nonpositive enabled value cap, duplicate/incomplete tiers, invalid dates, incompatible guest/account conditions, or a coupon code that is not a published WooCommerce coupon (or is a store credit or gift card code). An enabled ladder needs at least two completed levels. Campaign limits must be whole numbers from 0 through 1,000,000; a removed role must be reselected or cleared intentionally.

If an action expired, reopen the screen for a fresh form. If the promotion changed, compare your edits with its current version before resubmitting. If a save cannot be confirmed or the screen fails after an action, inspect the saved record first; repeating blindly can duplicate an intended operation. If license permission is required, resolve it on License. If storage is unavailable, retain the configuration and have the underlying WordPress/WooCommerce storage or campaign table repaired.

For preview, select an existing promotion and product, provide valid amounts and positive quantity, and use a valid local date/time. A before-start, ended, below-threshold or coupon-not-allowed result describes that sample, not a save failure. For basic coupons, use a unique code, supported type, nonnegative amount and a maximum spend no lower than its minimum; use WooCommerce's advanced editor for extension types.

The complete technical message lookup lists exact codes and messages, their conditions and recovery. It also distinguishes rule validation, coupon status, preview decisions and dependency failures.

How a promotion becomes a gift

Free Gifts & BOGO first checks which purchases count, calculates earned gift slots, resolves competing promotions, checks the real products available for those slots, and brings the cart's gift lines into agreement with the result. Earning a slot does not promise that a gift is still published, purchasable or in stock. Current product checks and checkout checks still apply.

Create the rule using the [promotion editor](#), choose its qualification and reward settings, then publish and enable it. The calculation needs a named, published, enabled rule and an active schedule. A new rule starts disabled with a zero threshold and no automatic product selected; those defaults are not a usable offer until completed. The [merchant schema reference](#) explains every saved field and its limits.

Only valid current cart facts count. Purchased product and variation identity, quantity, line amounts, product type, product traits and taxonomy membership come from the cart integration. Money uses the active currency's precision; quantity supports six decimal places internally. Duplicate cart-line identities, malformed amounts or inconsistent totals stop calculation rather than granting gifts from partial information. Taxes and shipping are not part of the engine's qualifying line amounts.

For developers building a local integration, the pure calculator takes normalized rules/settings, cart facts and an explicit time. It produces all rule decisions, selected gifts and suppressed gifts without changing the cart or reserving stock. Its results do not replace shopper, licensing, Core/B2B or checkout authorization. The candidate service must run after WooCommerce has initialized. Normal storefront use obtains those prerequisites through the plugin's [cart integration](#).

Amounts, quantities and repeated reward sets

A single rule awards a whole reward set for each complete qualifying threshold, up to its maximum sets per order. Multiply the awarded sets by the number of free items per set to obtain the number of gift slots. Partial thresholds do not award partial gifts.

Example	Gifts earned
Spend 50; receive two gifts; eligible spend 125; maximum sets 0	Two complete sets, four gifts. The remaining 25 does not form another set.
The same rule with maximum sets 1	One set, two gifts.
Buy three eligible items; receive one gift; eligible quantity 8.5; maximum sets 0	Two gifts.
Buy three; receive two gifts; quantity six; maximum sets 2	Two sets, four gifts.

Maximum reward sets per order defaults to 1. Use 1 for one award, a higher positive number to cap repetitions, or 0 to allow each full threshold to repeat. This limit counts sets, not the number of free products. It is separate from campaign-wide order/gift limits and account limits.

Money is rounded half up to the active currency's decimal places. For a currency with two decimals, 10.005 becomes 10.01. An amount so small that it rounds to zero cannot serve as a positive threshold or enabled maximum gift value. Quantities are rounded to six decimals before comparison: 2.9999995 becomes 3.000000. The engine uses checked integer arithmetic and refuses a calculation too large to represent; it never awards a partial result after an overflow.

Custom integrations must provide decimal strings or integers, not floating-point numbers, to the pure decimal helper. Negative amounts, exponent notation and malformed numbers are not accepted. There is no automatic currency conversion. Use the current store currency consistently throughout a calculation.

Which purchases count

Gift lines do not earn more gifts. For ordinary purchased lines, the plugin applies product-trait policies, exclusions and then the chosen purchase scope. A product must survive all exclusions even if it also appears in an inclusion list.

Choose **All products** to count every otherwise eligible purchase. Choose selected products/groups to use the union of your product, category, tag, product-type and attribute/grouping selections: a purchase needs to match any one selected group, not every group. Selecting a parent product also matches its concrete variations. An empty selected scope cannot qualify a purchase.

Sale, virtual and downloadable products each have a global allow setting, initially on. A rule can inherit that setting, explicitly allow the trait, or exclude it. Allowing one trait does not override another trait's exclusion or a product/category exclusion. A virtual downloadable sale product must pass all three applicable policies.

Gift cards sold with AffStackDev Store Credit are paid value, not merchandise. A gift card line counts toward no promotion, by spend or by quantity, unless **Gift cards** is ticked in [Settings](#); there is no per-promotion choice for it. A gift card product page therefore shows no promotion box while the setting is off. Paying with a gift card is a different matter: see [Qualification, minimum spend and coupons](#).

There are three distinct kinds of exclusion:

Exclusion	Effect
Purchases that do not count	Removes those product IDs or category families from eligible spend/quantity. The products may still be gifts.
Products that must not be free	Removes products or category families from gift choices. The products may still count as paid purchases.
Shared/global exclusions	Applies to both purchased-item qualification and gift eligibility. These include products, direct categories, tags, product types and attribute/grouping terms.

The scoped purchase and gift category exclusions include child categories. Category inclusions and older shared/global category exclusions match directly assigned categories. For example, an item assigned only to Shirts is excluded by a scoped Clothing-family exclusion when Clothing is its parent category. Merely selecting Clothing as an inclusion does not match that Shirts-only assignment unless Clothing is also directly assigned.

Variations use their parent product's category/tag/grouping assignments. Candidate gift type checks can recognize both the variation type and its variable parent type; the live purchased-item type qualification uses the catalog parent's type for a variation. These details matter when a product appears to be in a broad family but does not match the saved selectors.

See [qualification controls](#) and [eligibility controls](#) to change these settings. Storefront descriptions of a potentially relevant promotion do not by themselves prove that the actual cart has earned it or that a gift is available.

When a promotion is active

Leave a start or end blank for an open boundary. A promotion becomes active exactly at its start time and stops exactly at its end time. The end must be later than the start. For example, an offer ending at noon is no longer active at noon itself.

Editor dates use the site's timezone and are stored as UTC instants. The current stored interval controls availability. A future rule is before start; a past rule is ended. Malformed or reversed dates make the schedule invalid. A rule also needs to be published and enabled, so an active date window cannot activate a draft or disabled rule.

When a gift disappears at a boundary, review the actual rule's start/end and the site's displayed timezone before changing eligibility or stock settings. The plugin recalculates against current time; an earlier cart view does not preserve an expired offer.

Qualification, minimum spend and coupons

Single promotions qualify by eligible spending or eligible item quantity. Automatic rewards provide a specific product; customer-choice rewards offer an allowed pool; same-product rewards provide the exact product or variation bought. Same-product rewards require quantity qualification.

A quantity promotion can also require minimum eligible spending. This extra amount is a gate, not another repeating threshold. With buy three/get one and minimum spend 40, six qualifying items worth 39 earn no gift. Once the minimum is met, the quantity threshold determines the sets. The calculation applies the minimum whenever it is enabled, even though the editor presents it for quantity offers.

Coupon compatibility and spend basis are separate decisions. A rule can allow coupons, disallow them, or inherit the global allowance (initially allowed). It can calculate eligible spending before discounts, after discounts, or inherit the global basis (initially after). Before uses eligible line subtotals; after uses their discounted totals. Excluded products, taxes and shipping do not increase this amount.

Store credit and gift cards from AffStackDev Store Credit are payment, not discounts. They never reduce eligible spending on either basis and are not a coupon for coupon compatibility: a \$60 cart paid partly with \$20 of store credit still earns a spend-50 gift, while an ordinary coupon still counts as configured.

For a spend-100 offer, eligible subtotal 120 with discounted total 90 earns a set on the before basis and none on the after basis. If that rule disallows coupons, the current coupon/discount signal blocks it regardless of spend basis. The engine also recognizes a nonreward line whose total is less than its subtotal as a discount signal; custom integrations cannot hide that difference by sending a false coupon flag.

A promotion can also require one of its own coupon codes. Without one of them in the cart its decision is "coupon code required" (not qualified), whatever the cart's spend. Its own codes are exempt from its coupon compatibility: with a required code that takes 10% off, a promotion that disallows coupons still applies, and the code's discount lowers the eligible spend on the after basis like any coupon. When the cart lists its coupons, that list decides whether another coupon is present (a line discounted by the promotion's own code is not another coupon); a coupon the cart does not list (one made in code, without a coupon record) always counts as another coupon.

The displayed/calculated decision identifies the rule, qualification type, eligible totals, contributing lines, earned sets, any set cap applied, number of gifts, effective reward method, value basis and coupon basis. A rule may be inactive, invalid, below its threshold, or earned. Competing earned rules can still be suppressed by stacking. Product availability is checked after those calculations.

A spend or quantity threshold must be positive, and each award must contain at least one free item. An automatic reward needs a real chosen product or concrete variation. A selected gift pool needs at least one meaningful selector. Fix incomplete settings in the editor rather than relying on a stored draft to behave as an active promotion.

Reward levels: highest reached only

Use a reward ladder when spending or buying more should change the offer. Configure two to twenty completed levels with different positive thresholds. Each level has its own gift count, automatic product or customer-choice pool, and optional maximum gift value. Purchase scope, exclusions, schedule, minimum spend and stacking remain shared.

The shopper receives only the highest reached level. For levels spend 50 → one gift, 100 → two gifts, and 200 → three gifts, a qualifying spend of 150 grants the two-gift level. A spend of 450 still grants the three-gift level; it does not repeat the 200 threshold or add rewards from lower levels. The single-rule maximum reward sets setting does not change this highest-level behavior.

Normal tier configuration supports automatic gifts or customer choice. Same-product awards are configured as a single quantity rule. At a level, choose all eligible products or the level's selected pool; the single-rule option that mirrors purchase scope is not a separate stored tier pool option.

Changing cart contents can move the shopper between levels. The plugin rechecks the new level's product/pool and removes excess or incompatible gift lines. A previously chosen gift that does not belong to the new level cannot be retained just because it was available earlier.

Thresholds are compared at the currency's precision. Avoid distinct stored money amounts that round to the same effective threshold. If such values meet in the pure calculation, the stable tier identifier breaks the tie; the engine does not merge their rewards. Ordinary editor validation rejects duplicate stored thresholds and removes untouched blank level cards. See [tier controls](#).

Get the same product free

Same-product offers count each exact product or variation separately. Buying two blue variants can earn a blue gift; buying one blue and one red does not combine them for a buy-two offer. Two separate paid cart lines of the same variation do combine for its threshold.

For buy two/get one, quantities A=3 and B=3 earn one A and one B. They do not earn three gifts from a pooled quantity of six. The award is a separate free line; the existing paid merchandise stays paid. Gifts do not contribute to earning more same-product gifts.

The maximum reward sets setting caps all earned product groups together. If only one set may be awarded and more than one product qualifies, the plugin assigns it in ascending concrete product/variation ID, not by price or quantity. This makes the result stable but means the most expensive product is not automatically preferred.

Each awarded slot retains the exact product and variation. The live service checks stock for the rounded-up paid quantity plus the group's free items. For three paid units and one gift, the product must support four units under its WooCommerce stock/backorder policy. If the exact gift is unavailable, the plugin does not replace it with a different product. A separate promotion or explicitly chosen pool is needed when substitution is intended.

Combining promotions

Lower priority numbers run first; a lower rule ID breaks an equal-priority tie. A rule can be stackable, exclusive, or inherit the global setting. The global default is stackable.

Mode	What can coexist
Stackable	Other selected rules, unless a global exclusive rule already won. A stored group name alone does not make a stackable rule exclusive.
Exclusive with a group	Only the first earned exclusive rule in that same group wins. Other groups and stackable rules can coexist.
Exclusive without a group	Competes with every other rule. It wins only if nothing earlier has been selected; if it wins, every later earned rule is suppressed.

For a priority-5 stackable rule, priority-10 global exclusive rule, and priority-15 stackable rule, the two stackable rules survive. The exclusive rule loses because another rule was already selected. If the priority-5 rule is global exclusive instead, it suppresses both later rules. Exclusive groups Summer and Loyalty can each contribute their first eligible winner.

Qualification is calculated independently from the eligible paid cart, so the same purchased units may qualify more than one selected stackable promotion. A suppressed decision retains the winning rule's identity and the reason for the conflict. Customer targeting is applied before stacking, so an account-ineligible exclusive promotion cannot suppress an eligible alternative.

Gift availability is checked after stacking. If the selected winner's product is unavailable, the engine does not automatically return to a lower-priority suppressed promotion. Review the winner's product or deliberately change priority/group policy when this behavior is not the offer you intend.

Limiting the value of each gift

Enable a maximum reward value to restrict each gift's reference price. Choose current active price, regular price, or the global default (current). This is a per-product limit, not one shared budget across every gift in the award.

A gift priced 8 on sale with regular price 12 passes a maximum of 10 using current price and fails using regular price. Two 8-value gifts both pass that maximum; their combined value of 16 is not compared with 10. A gift exactly equal to the maximum is permitted.

The maximum must round to a positive amount in the active currency. The product must always have a valid current price. With a regular-price cap, a missing regular price makes the product unavailable for that gift; the plugin does not substitute current price. Publication, stock and all eligibility filters still apply.

Without a maximum, a valid current price is sufficient. In this release the recorded reference amount uses current price when the cap is off, even when the resolved basis label is regular. A later current-price change can refresh the gift's stored marker while leaving the same product selected. This reference is retained evidence for the gift, not a price charged to the shopper for the free line.

Choosing a gift pool

An automatic gift is one exact chosen product or variation. A variable parent is not a promise to choose any available variant: select the concrete variation. The resolver checks stock for all automatic gift slots together. Customer choice resolves one-unit candidates and lets the live cart layer validate the combined actual selections.

Pool	How it works
All eligible products	Reads the published catalog and applies all gift restrictions.
Selected/curated products and groups	Uses the union of selected products, categories, tags, product types and attributes/groupings. Matching any selected group is sufficient.
Specific products	Reads the selected products/variations; selecting a variable parent expands its concrete children.
Categories, tags or attributes/groupings	Uses matching products from the published catalog.
Same products or groups that count toward the deal	Mirrors the purchase inclusion scope. It does not require the gift to be one of the actual paid cart lines, and purchase-only exclusions do not become gift exclusions.

Where explicit product selections overlap catalog group selections, the explicit product observation is retained. This matters for hidden products: hidden gifts may be used when explicitly selected or fixed automatically, but are not offered merely because a catalog group would otherwise include them. Explicit selection does not bypass publication, purchasability, stock, exclusions or maximum value.

A valid empty pool means the current reads succeeded but no product passed the filters. An invalid pool means the plugin could not safely resolve its configuration or catalog facts. Both prevent unsafe gifts, but they require different recovery: review product/filter choices for an empty pool, and restore the failed configuration/API/taxonomy for an invalid read. Changing a rule while the shopper is choosing a gift requires a fresh entitlement; stale method, product, tier, pool or value settings are rejected.

Why a product can be rejected as a gift

A gift must have a coherent real product/variation identity and a usable name/current price. Product and parent must be published; variable/options-based products need a concrete choice; WooCommerce must consider it purchasable and in stock, with enough stock for the requested quantity. Backorders are permitted only under the product's actual stock policy.

The plugin then checks hidden-product discovery, sale/virtual/downloadable policy, gift-only and shared exclusions, any additional product-type restriction, membership in the selected pool, and the value limit. An explicitly selected product can still fail any of these checks. The first failure explains the rejection; fixing it may reveal another independent restriction.

A gift card is never a gift, whatever the settings: AffStackDev Store Credit accepts a gift card in the cart only with its purchase options. The editor refuses a gift card as the automatic gift, in named gift choices or in a same-product deal, and a promotion that already names one treats it as unavailable.

For curated pools and the mirrored qualification-scope pool, product types are one way to match the OR pool. For other pool modes, a nonempty reward product-type list further restricts the result. A variation can match variation or its parent variable type. Category and grouping facts come from the parent product; concrete price, stock, sale and downloadable/virtual facts come from the variation.

Custom candidate repositories must supply the complete normalized snapshot: identities and required quantity; name/SKU/image; discovery source, product/parent types and publication state; visibility and stock state; actual Boolean purchasability/stock/backorder/trait flags; integer current/optional regular prices; category/tag/grouping lists; and variation attributes. Candidate names/SKUs are bounded to 512 bytes and

attribute values to 200 bytes. Numeric or string substitutes for the required Boolean flags are rejected. Malformed identities, inconsistent stock flags or incomplete snapshots do not become gift permissions.

The exact snapshot schema and diagnostic names are in the technical reference at the corresponding candidate-contract and error sections. These are local integration contracts, not additional storefront entry fields.

Catalog reads and availability changes

The resolver reads current WooCommerce products. Catalog discovery is paged in groups of 100 and continues through the returned pages; the first 100 products are not a fixed total pool limit. Explicit product lists and variable children are deduplicated and processed in stable ID order. Missing products/children are reported instead of substituted.

For a variation, its parent provides categories, tags, grouping terms and catalog visibility. The variation provides its price, stock facts, image and concrete attributes. A missing or nonvariable parent makes that variation unusable. Registered taxonomies needed by the selected exclusions or groups must be readable. An unavailable taxonomy cannot simply be ignored, because that could bypass an intended exclusion.

Product availability is a current observation. The candidate list does not reserve stock, freeze prices or hold a gift indefinitely. Another purchase, a product edit, a price change or a promotion edit can change what is allowed before checkout. Review the refreshed cart and choose again when necessary. Current live checks are described under [selection](#) and [checkout revalidation](#).

Gift-line identity and retained information

Each gift line carries a server-created marker identifying the promotion, one-based gift slot, automatic/customer-choice source, concrete product/variation, and reference value/basis. Same-product gifts use the automatic source after their exact product is resolved. A marker does not contain a stock reservation or grant permission to keep an expired gift.

The plugin recognizes any use of its marker namespace, including damaged data. A malformed marker or one that disagrees with the surrounding cart product is scheduled for removal. The live cart can remove it only when ownership and line safety are confirmed. If safe cleanup cannot be confirmed, the line can remain on hold and block checkout until reviewed; it never becomes an ordinary purchased item that can earn more gifts. Do not edit markers or copy them to manufacture a free line.

Current candidate identity and reference information are compared with the retained marker. If they change, reconciliation can refresh the gift line. This protects the connection between the actual product, earned slot and recorded gift value. Order copies and retained attribution are explained under [checkout attribution](#).

What happens when the cart changes

The plugin compares current earned slots with existing gift lines. It schedules gifts for removal when entitlement is lost, a slot is no longer earned, a method changes, a product becomes unavailable, or the marker is invalid. Actual removal requires confirmed ownership and current line safety; an unconfirmed line can remain on hold for review. Every gift slot contains exactly one unit; editing its quantity does not earn additional free items.

Automatic gifts are added when missing and restored to the current permitted product. Duplicate lines for the same promotion/slot do not create extra awards: one current matching line survives. Same-product offers can

use different exact products for different slots. An unavailable automatic gift is removed without substituting an unrelated product.

For a customer-selected gift, the plugin keeps a valid current selection. If the same product remains eligible but its reference information changed, it can remove/re-add that same product with current information. If it leaves the pool or otherwise becomes unavailable, the slot becomes empty and the shopper chooses again. An untouched empty choice slot is never automatically filled with an arbitrary product.

For example, reducing an offer from three gifts to one removes the second and third slots. Switching to a higher or lower reward tier can also change the allowed pool. Removing and re-adding paid items may restore an entitlement, but the resulting gift must still pass current checks.

The reconciliation planner itself only describes removals, quantity corrections, additions and retained lines. The [live cart layer](#) performs those operations with its fresh-plan and mutation checks. Integrations should use that supported path rather than treating a plan or old marker as permanent authorization.

Guest, account and previous-order restrictions

A promotion can allow any shopper, guests only, or signed-in accounts. It can additionally require selected account roles or a particular paid-order history. Defaults impose none of these extra restrictions. Role/history restrictions require a signed-in account even when the access field says any. Guest-only promotions cannot combine with account-role, history or per-account order limits.

Selected roles use the shopper's current WordPress account roles; matching any selected role is sufficient. Public messages describe selected accounts without revealing role slugs. Changing an account's role can therefore change its eligibility at the next fresh evaluation.

History asks whether that account has at least one previous WooCommerce order with a paid date. It does not use billing email to attach guest purchases to an account, and it does not require a particular previous spend. Refunded or cancelled orders that retain a paid date still count. The current checkout order is excluded so it cannot satisfy its own previous-order requirement.

An account with no previous paid orders can qualify for that restriction only while no such history exists. Saving this restriction also sets a one-gift-order-per-account limit for that promotion. The campaign mechanism enforces this separate account allowance; see [campaign limits](#). A previous-paid-order restriction requires an actual prior paid order on the same account.

A failed account/history lookup does not mean the shopper is new or eligible. The promotion is withheld until current evidence can be confirmed. The plugin reuses facts only within one evaluation, then checks fresh eligibility during order creation, Store API checkout processing and pay-for-order continuation. A changed account, rule state or paid history can therefore stop an earlier gift from proceeding.

Account-ineligible promotions are filtered before stacking so they do not suppress an otherwise valid alternative. These customer conditions still cannot override a Core/B2B audience restriction, licensing rule, stock issue or campaign cap.

Understanding a missing or removed gift

Use the reported reason and the relevant settings to locate the boundary that failed. More than one condition may be wrong; the first visible reason is not a promise that every later condition already passed.

Situation	What to review
Promotion is inactive	Publication, enabled state, current start/end window and coupon compatibility.
No qualifying purchases or threshold not reached	Paid-item scope, purchase/shared exclusions, product traits, eligible spend basis, additional minimum spend and actual quantities.
An earned promotion loses to another	Priority, exclusive group and the winning rule's identity. An unavailable winner does not automatically activate a suppressed fallback.
Gift pool is empty	Current publication, concrete variation, purchasability, stock/backorders, hidden-product selection, gift/shared exclusions, pool union, type restrictions and maximum value.
Product was selected but is now rejected	Refresh the cart; the rule, tier, product identity, price, stock or pool may have changed. Choose a currently eligible gift.
Calculation or catalog read is invalid	Correct malformed saved/integration input or restore the Woo/taxonomy/account lookup. Partial results do not justify a gift.
Gift line is removed or reduced	Current entitlement/slot count, changed reward method, duplicate slot, stale marker or unavailable product. One unit belongs to each earned slot.
Sign-in requested	The promotion requires an account, role or paid-order history. Sign in to the intended account.
Guest-only message	Use guest checkout only when the offer and checkout support it; a signed-in account does not meet guest-only access.
Account not eligible	Current roles or paid-order history do not satisfy the rule. New-account offers still count refunded/cancelled orders retaining a paid date.
Eligibility could not be confirmed	Retry a fresh read or contact the store; do not assume a failed lookup grants eligibility.

Customer targeting uses these messages: “Sign in to check eligibility for this gift promotion”; “This gift promotion is available to shoppers checking out as guests”; “This gift promotion is not available for this customer account”; “This gift promotion requires an account with no previous paid orders”; “This gift promotion requires a previous paid order on this account”; or “Customer eligibility for this gift could not be confirmed. Please try again or contact the store.”

Developers and support staff can use the technical reference's exact engine error lookup, including input field patterns, candidate rejection codes, stacking reasons, reconciliation removals and `afsd_fp_customer_*` account errors. Shopper-facing request and checkout recovery is also documented under [checkout errors](#).

How gifts reach the cart

Free Gifts & BOGO adds each gift as its own WooCommerce cart line with quantity 1 and a zero product price. Paid copies of the same product keep their normal price. A gift still follows the product's ordinary shipping and stock requirements; a zero product price does not itself make delivery free.

Automatic promotions add eligible gifts when the cart qualifies. Customer-choice promotions create one choice for each earned gift slot. The same current promotion rules are used by classic carts and Cart/Checkout Blocks.

Qualification can change when purchased items, quantities, coupons, account eligibility, campaign allowance or available gift products change.

A current [license permission](#) and [consumer eligibility decision](#) are needed for live gifts. If they become unavailable, the plugin tries to remove gifts that it can safely identify as its own. When a line's ownership or protection cannot be confirmed, checkout can remain on hold until the cart is reviewed. Paid merchandise is not deliberately converted into a gift to bypass that check.

Gifts that match the purchased product

A same-product promotion adds separate free copies of the qualifying product or exact variation. It checks enough stock for both the qualifying quantity and the earned gifts. For example, 2.25 qualifying units and two gifts require stock validation for five units: the purchased amount rounded up to three, plus the two gifts. Only two free lines are added. The order describes these gifts as **Added automatically**.

What the cart uses to qualify

The plugin reads WooCommerce's calculated line amounts, quantities, products, variations and coupon presence. A rule can use the applicable amount before or after coupons, as configured in the [promotion rules](#). Store credit and gift cards are treated as payment: their part of a line is added back and they do not count as a coupon. Product type, sale/virtual/downloadable flags and selected catalog categories, tags or other taxonomies can affect eligibility. Gift lines do not earn further gifts as ordinary purchased merchandise.

A missing product, missing referenced taxonomy or invalid amount can prevent a reliable gift calculation. Reload the cart after correcting the underlying product or cart problem. The merchant preview uses the same product/amount normalization, but its simulated product and entered totals do not reserve campaign allowance or establish the current shopper's eligibility; see [previewing a promotion](#).

Recalculation and changed carts

The plugin rechecks gifts after WooCommerce calculates cart totals. It removes obsolete gifts, keeps valid choices, restores each gift quantity to one and adds earned automatic gifts. When a change occurs, it runs one additional guarded totals calculation so the cart reflects the updated gift lines. A restored cart session is recalculated even when WooCommerce has cached totals.

Gift entitlement is checked again around cart changes. If another calculation changes the purchased items, their amounts or the shopper's eligibility during a gift update, the plugin stops that update and asks the cart to refresh. This protects against saving a choice based on a cart that no longer exists.

An unavailable or stale choice may disappear. A choice for the same still-eligible product can be refreshed when its saved gift details have changed. If the gift cannot be safely removed or replaced, review the cart before checkout rather than repeatedly submitting the same change. The plugin's checks do not override another product's protected cart line.

Choosing and changing a gift

Each earned choice slot holds one eligible product or variation. Select from the currently offered list; the server rechecks that product when saving. A choice displayed earlier may no longer be available after a stock, cart or promotion change.

Gifts in customer-choice slots are optional. Shoppers can leave them unselected and continue to checkout. Choosing the product already saved in a slot leaves the existing choice intact. Changing a gift removes the earlier slot choice and attempts to add the replacement. If adding fails, the plugin tries to restore the previous item while the cart still qualifies, but the resulting cart must be reviewed because restoration is not guaranteed.

Removing an automatic gift is not a saved opt-out: while its promotion still qualifies, a later cart calculation can add it again. A removed customer-choice gift can stay unselected.

Using a classic cart

A classic cart shows eligible choice promotions before the cart totals area. For each **Free gift** slot, choose a product from the list and press **Save free gift**. The page returns to the cart and confirms the saved choice, or explains that it is no longer available. If there are no eligible candidates, the section says so.

The form works without JavaScript. An expired form asks the shopper to try again; reload the cart to obtain a fresh form before saving. An unavailable cart also requires a reload. Gift quantity is fixed at 1, so editing it through the ordinary cart quantity control is not supported.

Automatic cart notices explain when an earned gift was added, when a gift was removed after a qualification or availability change, and when a gift could not be validated. These messages are standard WooCommerce notices.

Store API behavior for shopping integrations

Cart and Checkout Blocks use WooCommerce's authenticated cart updates. The browser sends only the intended promotion, slot and product identifiers; the server supplies and validates the gift identity. A client cannot make an ordinary product free by supplying a gift marker in a normal Store API add-to-cart request.

A rejected selection leaves the shopper with a current-cart recovery step: refresh, inspect available choices and choose again. It does not grant the earlier selection merely because it was once shown. The integration does not require a store administrator's permission for a shopper's ordinary authorized cart action.

Information shown to the shopper

The response provides the merchant's gift label, current validity, kept gift count (and how many of those gifts have a reduced price), unselected earned slots, current choice groups, product names with each choice's price (**Free** or, for a gift at a discount, its price and discount), exact variation labels, optional thumbnails and progress messages. Each gift line also says **Free gift** or its discount (**Gift at 50% off**). It does not expose internal authority proofs or private gift marker arrays. The gift count is the number of valid kept gift lines, not the count of every raw or suspicious cart entry.

A selection starts with one slot and one candidate. Quantity is locked to one in Blocks, including its minimum, maximum and quantity step. A cart refresh can change the list, pending-choice count and selected products together. Progress and the final gift choices are different displays; follow the available current choice controls when adding a gift.

Using the Cart and Checkout Blocks picker

The **Free Gifts** section uses the merchant's configured customer label. Each slot shows **Gift 1 of 2**, for example. Choose a product card, review **Ready to add**, then press **Add gift**. The radio selection alone does not add a

product. A saved gift shows its name, variation and **Added • Free** (for a gift at a discount, **Added •** with its price, such as **Added • \$4.99 (50% off)**, and each choice card shows that price instead of **Free**).

Use **Change gift** to open alternatives and **Update gift** to save a replacement. **Cancel change** closes the alternatives without saving the draft. **Remove gift** removes that selected choice through WooCommerce's ordinary cart action. You can then choose another or continue without it.

When more than eight candidates are available, **Search available gifts** filters names and variation descriptions, and **Previous/Next** move through eight choices per page. Search is local to the available list. No matching search results and no eligible gifts are different states; try another name for the former, or continue shopping/check out without a choice for the latter.

Controls pause while the cart is recalculating, coupons or shipping are changing, another item is being changed, or checkout is processing. Only one gift change runs at a time across the displayed picker. This component is independent of the merchant's optional enhanced administration display.

The picker checks the refreshed cart before confirming a change. If the network response is uncertain, it refreshes once and asks you to review what is now in the cart; it does not automatically repeat the attempted change. **Refresh gifts** requests current cart information. If that request fails too, **Reload cart** becomes available. Check your connection and reload before another attempt.

The section reports when changed cart contents remove a previous choice, how many gift slots remain open and how many gifts are in the cart. At checkout it can also show promotion progress and a **Promotion details** link opening in a new tab. The separate cart progress area is described under promotion progress. Product cards support keyboard selection, visible focus, live status announcements and layouts that adapt to narrow screens.

Gift records on an order

The order stores the gift's promotion, slot, product or exact variation, selection source and reference value. The promotion and item names are saved at checkout, so later renaming a campaign does not rewrite those labels on existing order evidence. The reference value is a gift valuation for reporting; it is not money paid by the shopper or a cash refund entitlement.

In WooCommerce order administration, each gift shows **Free reward from:** (for a gift at a discount, **Gift at 50% off from:** with its discount) followed by the saved promotion name, plus **Customer selected** or **Added automatically**. The order also records a gift's price setting and the price it was charged; the customer's order summary adds the discount to such a gift's name ("Socks at 50% off"). The **Gift promotions** order section groups the gift names by promotion; a promotion that gave the order its free shipping is listed there with "Free shipping". The same saved summary appears in order emails and customer order details, using the merchant's current customer-facing gift heading.

The summary reports attributed gift lines. It does not prove that an old order would qualify under today's rules. Historical referral-offer information has its own saved evidence; missing historical evidence is not filled in from the current campaign.

Account changes during checkout

A gift order must belong to the current shopper when checkout is submitted. The plugin checks both the working order and its saved customer identity. A signed-in shopper may take over the specific guest checkout draft associated with their Woo session, but an order saved to another customer's account cannot be silently transferred with its gifts.

If checkout asks you to refresh and review gifts after an account or cart change, return to the cart under the intended account, reload and inspect the choices before submitting again. Merchants should preserve the order's existing customer and gift records while investigating an identity mismatch. Changing current eligibility does not erase historical gift attribution.

Campaign and customer allowances

A promotion can limit the total number of campaign orders, the number of orders per signed-in customer and the total number of gift units. Zero means unlimited for that dimension. A promotion that gives free shipping uses one gift unit for each order it gives free shipping to. A positive per-customer order limit requires signing in; a guest email address is not a substitute for the account identity.

These allowances are separate from the per-order gift cap and the product's stock. For example, a campaign with ten gift units remaining cannot admit an order requiring eleven gifts even if the product is in stock. A campaign that has reached its allowance is excluded before choosing between competing promotions, allowing another eligible promotion to be considered.

The cart shows current availability, but another shopper can complete checkout first. Actual checkout reserves the allowance under a database lock and can ask the later shopper to refresh if the cap has been reached. Repeatedly reloading does not increase the allowance. A temporary busy message means another checkout is updating it; wait briefly and try again.

What remains reserved

At checkout, limited campaigns record an order-specific reservation. Both reserved and consumed orders count against campaign usage. The reservation belongs to the order and its saved customer, not simply to the browser that is currently open.

Reservations do not expire on a timer. Review abandoned or failed orders through their normal WooCommerce lifecycle rather than assuming allowance will automatically return after a few minutes. The campaign records remain available for payment retries and late confirmations, including after uninstall/reinstall. Deleting campaigns or settings does not clear those retained order-allocation records.

An error after saving a reservation may need merchant review even if the shopper saw a failure. Check the actual order and payment state before increasing caps or attempting to recreate the allowance.

Payment retries, cancellation and refunds

A valid reservation is honored during delayed payment confirmation even if the merchant subsequently tightens the campaign limit. The plugin confirms the original customer, exact gift allocation and saved reservation. It does not use the current visitor's identity to consume a delayed payment's allowance.

An unpaid order changed to **Failed** or **Cancelled** can release an unconsumed reservation. A paid order does not receive that release, and a consumed allocation is not turned back into unused allowance. A released order attempting payment again must meet the relevant current policy and available allowance. A released new-customer offer cannot revive if another paid order has since made the customer ineligible.

Refunds do not restore consumed campaign allowance. Restocking a gift through WooCommerce and returning campaign allowance are different actions. If an allowance cannot be confirmed during a payment or status transition, the merchant receives an order note; check the payment gateway outcome and gift availability before fulfillment. If a release cannot be verified, its note says that allowance remains reserved.

Stock and refund review

Gift products use WooCommerce inventory. When a reward order fails, Free Gifts & BOGO asks WooCommerce to restore or release stock using its normal order stock handling. It does not maintain an alternative product stock balance.

The order's **Free Gifts & BOGO inventory** panel describes the current Woo records: unmanaged stock, fully or partially reduced stock, inventory restocked by a refund, released stock or stock not yet reduced. These descriptions reflect stored Woo metadata. They do not themselves perform another stock movement.

The refund-review setting is enabled by default. If a refund leaves at least one gift while refunding a paid-item quantity or a positive amount, the plugin can add a merchant note asking for manual promotion review. Review the remaining paid and free quantities and the store's policy. The plugin does not automatically remove gifts, change the cash refund or charge the shopper for a retained gift.

A gift-only refund with no money amount does not by itself trigger that review condition. Refunding all gift units removes the remaining-gift condition. Order administration can show a current refund-review warning based on the order's refunds even when adding new review notes is disabled. The saved note is added only once for each refund that receives it successfully.

Where lifecycle information appears

Promotion names and gift names appear beneath the order table in emails and customer order details. Merchant order screens also show the attribution, stock summary and any refund warning. Checkout and payment processing perform the reservation checks; these historical displays do not re-evaluate the shopper against today's campaign to rewrite an older order.

Checkout messages and recovery

What the shopper or merchant sees	What to do
The gift selection expired	Reload the classic cart and submit its fresh form.
Cart unavailable or could not be validated	Reload the cart. If it continues, the merchant should inspect missing products, taxonomy configuration and current integration status.
A gift is no longer earned or a product is not eligible	Review changed cart contents, stock and current choices. Select only a currently offered gift.
A gift could not be added or the change could not be confirmed	Inspect the refreshed cart before retrying; the earlier request may have changed it.
Gift options could not be checked	Use Refresh gifts ; if that also fails, check the connection and use Reload cart .
A free gift could not be verified safely	Review current account eligibility and the integration hold . Preserve uncertain protected lines for merchant investigation.
Sign in to use a promotion with a per-customer limit	Sign in to the intended account, then refresh its cart.
Another checkout is updating gift availability	Retry after the concurrent checkout finishes.

What the shopper or merchant sees	What to do
Campaign or customer limit reached	Refresh to see available gifts or another qualifying promotion.
Saved allocation no longer matches the order	Contact the merchant before retrying payment; preserve the order and its gift records.
Allowance could not be confirmed for payment/status change	Merchant: check gateway payment state and gift availability before fulfillment.
Allowance remains reserved because release was not verified	Merchant: resolve the retained reservation before increasing the campaign limit.
Refund review needed	Merchant: review paid/free quantities and the store's return policy; no automatic clawback has occurred.

API selection failures

A shopping integration receives a specific rejection when the request is invalid, the requested action is unsupported, the cart is unavailable, a slot is invalid or its product is no longer eligible. The shopper should refresh and select from the current list. The technical reference lists the exact codes for support.

Confirmation and status labels

Your free gift selection was saved and **Your gift is now in the cart** are selection confirmations. **Gift removed.** **You can choose another or continue without it** confirms that the selected choice no longer appears in the returned cart. **Your cart changed** means the displayed gifts have been refreshed to match current conditions. A temporary **Updating cart** or **Updating your gifts** state disables conflicting changes until WooCommerce finishes.

The master technical reference contains the exact API codes and complete current message lookup for integrations and support.

Choose where shoppers see promotions

Free Gifts & BOGO can show current offers on a policy page, relevant product pages, catalog listings and the cart. These displays explain the offer. Actual rewards still depend on the current cart, shopper eligibility, available products and campaign allowances at checkout.

Open the promotion's [display controls](#) and choose each surface separately. A new promotion shows on product pages, in the cart and on the policy page by default; the catalog badge starts off. Turn a surface off when you do not want its message shown. This does not disable the promotion itself or prevent a qualifying cart from receiving its reward.

For an offer to appear, the saved promotion must be published, enabled, inside its active schedule and enabled for that particular surface. Current product-license and commerce eligibility must also permit presentation. If several promotions apply, they are presented by lower priority first, then by their saved ID. The display order is not a guarantee that every promotion will be awarded together; [stacking rules](#) determine the actual combination.

Descriptions and cards use the theme's normal text color. Cards wrap to narrower screens, and product disclosures can be opened with keyboard controls. Links have visible focus styling. Catalog badges are compact

green labels. Cart progress uses an ordinary progress bar and readable message, so the number alone is not needed to understand the remaining requirement.

Publish a Current Promotions page

Create or edit a normal WordPress page for your promotion terms. Select that published page as the policy page in [settings](#). You can write your store's terms above the current offer list.

Choose either of these placements:

Placement	What to do
Automatic list on the selected policy page	Leave the page without an explicit Free Gifts & BOGO list. The plugin appends the current promotions after your page content.
WordPress block	Insert Current Free Gifts & BOGO Promotions . The editor shows a placeholder; the live page displays the current saved promotions. Use Show Current Promotions heading to show or hide the heading. Wide and full alignment are supported.
Shortcode	Insert <code>[freebie_promotions_current]</code> . Use <code>[freebie_promotions_current heading="no"]</code> to hide the heading.

An explicit block or shortcode on the selected policy page prevents the automatic extra list. If you deliberately add two explicit placements, both can appear. The block and shortcode can also be used on other appropriate content pages; selecting the policy page controls the automatic append and the details links.

The list heading is **Current Promotions**. Each card shows the promotion name, its main requirement and reward, and any short description. Tiered promotions also show each configured reward level in order. The full promotion description appears below the summary and can contain supported page blocks and formatted text. Customer targeting and campaign-limit notices are added to the short requirement when configured.

When there are no current offers for this request, the section says **There are no current free gift promotions**. Hiding the heading does not hide this message. The global **Customer-facing label**, initially **Free Gifts**, supplies the section's accessible label; it does not rename every built-in heading.

Keep the selected policy page published. Product and cart details links disappear if it is missing, is not a page or is not published. A published page's own access or password settings still govern what a visitor can read after following its link.

Read product offers and catalog badges

On a relevant product page, **Free gift offer available** opens a disclosure with the matching promotion names, requirements and reward summaries. Short descriptions appear there when supplied. **View promotion details** links to the published policy page for fuller terms and individual tier levels.

The optional catalog badge says **Free gift offer**. Its accessible label identifies the first matching promotion. It does not show a reward count or mean the shopper has already qualified. Both classic Woo product layouts and Woo product-price blocks have supported placements; the plugin avoids repeating its normal block/catalog output for the same product within a page request.

Product relevance uses the displayed product's catalog facts and the promotion's qualification scope and exclusions. Variations use their parent's catalog categories and tags with the concrete variation ID. A variable

parent can also be shown when a specifically selected child variation qualifies. This display check does not establish that every variation qualifies, that a purchase threshold has been reached, or that the intended free product is currently available.

For example, a product can show an offer while a shopper still needs more qualifying items, a different eligible variation, a signed-in account, or available campaign capacity. Choose the exact variation and inspect the current cart before relying on the reward. If an ancestor category is excluded for qualifying purchases, a product-page hint can remain even though the live cart excludes a product assigned to its child category. Use the current cart result for actual qualification. The [qualification rules](#) explain what counts toward the offer.

Interpret reward summaries

Quantity summaries say **Buy N qualifying items**; spend summaries say **Spend AMOUNT** in the store's current money format. The summary is brief: it does not list every scope exclusion or extra minimum spend, so keep full terms accurate.

Reward wording follows the configured method:

Method	Shopper summary
Customer choice	Choose the stated number of free items. The actual eligible choices come from the cart's reward picker.
Same product	Receive the stated number of additional free units of the same product, subject to the same-product rules.
Automatic product	Receive the stated quantity of the named free product or variation. If that catalog record is missing, the summary can fall back to “a free product”; the missing reward still needs correction.
Tiered reward	The highest reward level reached applies. The policy card lists the individual levels.

Limited campaigns state that gift availability is confirmed at checkout. An account order limit adds a sign-in requirement and the maximum orders per account. These notices do not display a live remaining allowance and do not reserve a gift merely because the page was opened.

Read cart progress and remaining requirements

The **Free gift progress** section appears before the classic cart table or above the Cart block. It uses the current cart's evaluated qualifying spend or quantity. An optional **Details** link opens the published policy page.

Before the first level, the message says how much more qualifying spend or quantity is needed. After one level is unlocked, it can show the amount needed for the next reward level. At the highest level, it says the promotion is unlocked. The bar targets the next unmet level, or the highest level when all have been reached, and does not exceed 100%.

For a promotion with quantity levels at three and five items, a qualifying quantity of three unlocks the first level but still targets five for the next bar. If the rule also has an additional minimum spend that has not been met, the message switches to that spend requirement. Reaching the quantity alone does not bypass the extra minimum.

“Unlocked” describes the displayed threshold progress. The cart must still have eligible reward candidates, valid customer access, available campaign capacity and a successful final checkout decision. A product card or progress bar is not a reserved gift.

Some holds are displayed instead of ordinary progress:

Message or condition	What to do
Sign in to qualify for the limited promotion	Sign in to the intended customer account, then refresh the cart.
Campaign or account limit reached for the gifts in this cart	Review the promotion's allowance and the current cart. Removing a display message does not restore an exhausted allowance.
Gift availability could not be confirmed	Retry after the relevant availability or campaign problem has been resolved.
Customer eligibility or purchase-history message	Follow the stated account/history requirement; see customer targeting .
Coupon combination not allowed	Ordinary progress is omitted for that decision. Remove the conflicting coupon or use a promotion that permits the combination.
A promotion that needs a coupon code	Nothing is shown for it until one of its codes is applied; then its ordinary progress appears.

No progress can also mean there is no valid current cart evaluation, the promotion's cart display is off, the rule is outside its active schedule, or its decision is suppressed by another rule. Do not infer that a missing bar means the stored promotion was deleted.

For the Cart block, the initial page includes a server-rendered fallback. The browser then reads the Woo Cart Store's server-provided progress and updates the visible messages when that data changes. It does not independently calculate qualification or choose gifts. A valid empty progress list clears the section's contents; an unavailable Cart Store can leave the original fallback visible. Refresh the cart/page when it no longer reflects the current cart.

Product disclosures, catalog badges and the policy page do not have this Cart Store refresh. They reflect the request that rendered the page. Cached page HTML is not a fresh eligibility decision for a changed shopper or cart. Check the actual cart after changing products, account, coupon or quantity.

Preview a saved promotion before enabling it

Use [Promotion Preview](#) to examine a saved promotion without changing inventory, its enabled state, a shopper cart or a customer session. Store staff need WooCommerce management access to use the form.

1. Save the promotion first. Preview uses its saved version, not unsaved editor changes.
2. Choose **Promotion to test** and **Product in the sample cart**. Select the exact variation when options affect eligibility.
3. Enter **Cart value before discounts** and **Cart value after discounts** for the entire sample product line, not its unit price. The after-discount value must not exceed the before-discount value.
4. Set **Quantity in cart** to a positive quantity. Use a whole number for ordinary products.
5. Choose **Date and time to test** in the store timezone. Type the sample cart's coupon codes under **Coupon codes applied**, and set **Another coupon is applied** if the simulated cart should also be treated as containing a coupon that is not listed.
6. Select **Evaluate Promotion**, then read the result and the saved promotion status separately.

Preview evaluates a draft or disabled rule as if it were published and enabled, but leaves its real saved state unchanged. Its schedule, product scope, exclusions, coupon policy, reward levels and evaluator limits still apply. This lets you inspect an upcoming promotion at a future time without publishing it now.

The coupon checkbox indicates coupon presence. It does not apply a coupon code or calculate the discounted value for you; enter the intended whole-line amounts. The preview uses one product line, so it cannot represent every mixed-cart or overlapping-promotion scenario.

Reward earned in this preview means the sample input earned the rule's calculated reward. **No reward in this preview** can be a valid result, such as being below a threshold, outside the chosen schedule or using an excluded product. **Preview needs attention** means setup or input could not be evaluated. Review qualifying spend, qualifying quantity, reward and any selected reward level shown on the result.

A successful preview does not verify the real shopper's role, purchase history, account eligibility or campaign allowance. It does not reserve capacity or complete the cart's candidate-selection and reward-creation process. The live cart and checkout repeat their own checks. When you are satisfied with the saved setup, return to the promotion editor and enable/publish it through the normal controls.

If the selected promotion was removed or trashed, choose an existing saved promotion. If the product or variation no longer exists, choose a current item. For an incomplete or invalid rule, open it, correct the highlighted fields and save before previewing again. The form's promotion selector shows at most 500 alphabetically listed promotions; open another promotion directly and use **Preview saved version** when it falls outside that list.

Understand promotion reports

Open the [dashboard or reports](#) to compare recorded promotion use. Choose **7 days**, **30 days**, **90 days** or **365 days**; the default is 30. These are rolling windows measured back from the current UTC time by order creation date, not complete local calendar months. They do not filter by the date a refund happened.

Reports include Woo orders currently in **Processing**, **Completed** or **On hold** with usable Free Gifts & BOGO reward attribution on their order lines. Pending, failed, cancelled and refunded statuses are excluded. A promotion does not need to remain active or even present in the current editor for its saved order attribution to contribute. Its displayed report status describes its current saved state, separately from the historical order use.

The report reads existing Woo orders. It does not create a second financial ledger, issue rewards, change an order, repair missing attribution or reconstruct usage before valid attribution was recorded. A gift added outside this plugin's attribution path is not automatically treated as a Free Gifts & BOGO reward.

Read each report value

Value	Meaning
Promotion orders	Unique included orders with at least one usable attributed reward. An order using several promotions is counted once in this overall total.
Gift items issued	Current attributed order-line quantities across included orders. This is not a separate count of shipments, selections, refunds or customer visits.
Promotion-order value	The whole value of each included order, less its recorded refunds, never below zero. An order is included once in this total.

Value	Meaning
Promotion comparison orders	Included orders using that particular promotion. One order can appear under several stacked promotions.
Attributed order value	The full net value of orders using that promotion. Values overlap between stacked promotions and must not be added together as a unique sales total.
Average order	That promotion's attributed order value divided by its included order count, rounded to the report's money unit.
Most-used reward	The highest-quantity reward recorded for that promotion.
What customers received	Reward quantities grouped by concrete product/variation identity, or by saved reward name when no positive product ID is present.
Recent promotion orders	Up to ten most recently created included orders, with their current recorded gift-item quantity and net order value. Follow the order link to inspect the native record.

The current leader is ranked first by order count, then by gift-item quantity, then by attributed order value. Ties use name and saved ID. It is not necessarily the promotion with the highest revenue or greatest financial benefit. Reward lists rank by quantity, then name and product ID.

Order value is the entire order's remaining total, including tax, shipping or other components already in that Woo total. It is not the retail value of the gifts, discount savings, profit or proof of payment. On-hold orders are included, so the report should not be read as a settled-payments report.

A promotion's report name comes from retained order attribution and can differ from its current editor title. Reward names can likewise reflect saved history. The report counts actual line quantities; changing an order's quantity or attribution can change a later report. Inspect the underlying order when names or quantities need explanation.

Interpret refunds and currencies

A recorded refund reduces the reported order value while the order remains in an included status. It does not subtract returned gift quantities from **Gift items issued**. If the order is moved to **Refunded**, the entire order drops out of this report because that status is excluded. If it remains in an included status with zero net value, it can still count as a promotion order with reward units.

For example, an included order worth 100 with a recorded refund of 25 contributes 75 to the overall promotion-order value. If two promotions supplied rewards on that order, each promotion can show 75 in attributed order value, while the overall total still receives only 75. A returned free item's quantity is not automatically subtracted from the free-item count by this aggregation.

Reporting follows order creation time. A refund recorded today can alter the value of an older order in a 90-day report even when it does not appear in a 7-day report. The date of the refund does not bring an old order into a shorter creation-date window.

The report does not separate historical order currencies or convert exchange rates. Its displayed money uses the store's current Woo currency formatting. If the store contains orders in different currencies, inspect their native order currency values before relying on the combined monetary figure. Product quantities can still be understood separately from that monetary limitation.

Recognize partial results and cached reports

The normal scan reads up to 50 pages of 100 Woo orders, and up to 5,000 matching promotion orders. Technical integrations can alter the supported bounds, but the report remains bounded. Ordinary orders without Free Gifts & BOGO attribution still consume scanned pages. A large store can therefore reach the page limit even when relatively few matching promotions are shown.

When the report warns that it is a partial result, choose a shorter reporting window before comparing totals. The full report displays only the first 250 ranked promotions and rewards, and up to ten recent orders. The compact dashboard comparison shows the first five promotions. A list with more than 250 reward products alone can be capped without the larger-result warning, which is triggered by the number of promotions. These visible-row caps are separate from the scan limit and overall aggregate values. There is no native report CSV or file-export button, despite export wording in the large-result notice.

An individual order is also limited to its first 500 line items for this report. Further lines are omitted from attribution reconstruction. This condition has an integration notification but does not set the normal page-limit warning, so investigate an unusually large order in Woo directly.

Reports are cached for up to five minutes in separate entries for each supported window. Checkout creation, relevant order updates, status changes, refunds, deletion and trash events clear those caches. A simple browser reload can still reuse the same report during its cache lifetime when no invalidating event occurred. Changes that do not emit one of the observed events appear after the cache expires.

If reporting is temporarily unavailable, check Woo order storage and try again. A reporting failure does not change saved promotion rules or checkout behavior. Keep native order records as the source for financial and fulfillment details; the report is a bounded view of their current attributed promotion use.

Resolve display preview and report problems

Problem	Recovery
Current Promotions is empty	Check that the intended promotion is published, enabled, active now and enabled for the policy surface. Check current license and commerce access.
Product offer or catalog badge is absent	Check its separate display flag, scope and exclusions, including the selected variation. The catalog badge begins off for a new rule.
Details link is absent	Select an existing published WordPress page as the policy page.
Offer appears but no free reward is added	Review actual cart qualification, candidate availability, customer targeting, coupon combinations and campaign limits. Presentation alone does not reserve or issue a reward.
Progress requires spend after quantity was reached	The additional minimum spend is still unmet. Add qualifying spend or correct the rule if that minimum was unintended.
Cart-block progress appears stale	Refresh the cart/page and check the Woo Cart Store integration. Product/policy HTML has no equivalent live progress refresh.
Preview cannot find the promotion or product	Select an existing saved promotion and current product or exact variation; correct and save invalid fields before retrying.

Problem	Recovery
Preview earns a reward but a real shopper does not	Compare the real account, purchase history, campaign allowance, cart composition and current time. Preview supplies only its one-line hypothetical input.
Promotion totals seem to exceed the overall sales total	Stacked promotion rows each receive the whole order's net value. Use the unique overall order total rather than summing overlapping promotion rows.
Refunded gift quantity still appears	Gift quantities are not reduced by refund-line quantities in this report. Inspect the order and its current status.
Report seems incomplete	Check the period, included statuses, valid line attribution, query warning, 250-row display limit and 500-line per-order limit.
Recent changes are not reflected	Allow the five-minute report cache to expire when no observed order event invalidated it. Reloading alone does not force a fresh scan.

The [merchant controls](#), [promotion preview](#), [engine errors](#) and [checkout normalization](#) provide the related settings and precise input requirements.

License status and promotion activity

Free Gifts & BOGO uses its own product license. Activate the Free Gifts & BOGO key before running promotions or making promotion changes. There is no introductory operating period without an accepted license, and entering this key does not activate Suite or create a Suite key.

The product keeps two separate answers: whether its own license is currently accepted, and whether new promotion activity is allowed. AffStackDev's current plans use independent product entitlement and do not require a Suite license; an older plan that explicitly requires one also needs current verified Suite permission. Server restrictions can pause new activity even while the Free Gifts & BOGO license itself is shown as **Licensed**.

The license page therefore shows both **License status** and **Promotion activity**. Read the **Suite** row when a required companion entitlement is involved. A license grants no extra WordPress role, shopper eligibility, stock allowance or campaign capacity.

A hold preserves saved promotions, settings, reports and order history. It does not undo accepted orders, WooCommerce payments or refunds. Current cart and checkout activity still needs fresh permission; an old promotion or preview result is not enough to apply a new free gift.

What remains available during a hold

You can review saved promotions and reports, run the read-only preview against a saved promotion, and use license recovery, subject to your normal store-management permissions. Preview shows calculations; it does not grant shopper eligibility or reserve stock or campaign allowances.

Creating or changing promotions, duplicating them, changing their operating settings, creating a promotion policy page and saving or acting on coupons through the product require current promotion permission. Bulk promotion changes skip the requested records during a hold. The plugin also protects its rule storage when another caller uses WordPress's post or metadata APIs.

Three preferences can still change through the ordinary PHP settings form: the administration display mode, deletion on uninstall, and refund-review warning. Other promotion settings remain at their saved values. Suite's in-place settings-save action pauses with other mutation actions during a hold, so use the ordinary settings form for these permitted preferences.

Live cart activity stops relying on the unavailable promotion permission. The plugin removes only gifts it can positively identify as its own and safely change. Checkout may ask for a refreshed cart or hold an unresolved gift instead of accepting a stale offer. See [cart revalidation](#) and [campaign limits](#) for current-checkout versus already accepted order behavior. Existing order history, stock/refund processing and admitted campaign reservations have their own retained evidence; a later licensing hold does not cancel them automatically.

Server read-only, lock and stop restrictions all pause new promotion activity. Protected plugin updates have a separate readiness check based on Free Gifts & BOGO's own accepted product entitlement, installation credential and signature support. A required-Suite or activity-policy hold alone does not necessarily stop downloading a protected product update.

When your plan requires Suite

Suite is optional, needs no key of its own, and AffStackDev's current plans do not require a Suite license; this section applies only to an older plan that explicitly requires one. Installing Suite can organize the product's navigation without changing a plan into a Suite-dependent plan. If your accepted plan explicitly requires a Suite license, Free Gifts & BOGO checks Suite's own current public license status each time it evaluates permission.

The **Suite** row on the license screen explains whether the required entitlement is verified, a compatible public reader is needed, or the entitlement is not currently verified. With an independent plan, it instead says that compatible Suite is active or optional Suite is absent and independent product entitlement applies.

For a required plan:

1. Ensure a compatible AffStackDev Suite with public license contract 1 is active. The recovery message names Suite 0.5.0.5 or a compatible release with that reader.
2. Open Suite's **License** screen and recover Suite's own license there.
3. Return to Free Gifts & BOGO's license screen and review **Promotion activity**.

Free Gifts & BOGO does not copy, store or manufacture a Suite entitlement. Its reader requires fresh Suite evidence and a future permission deadline; a previously displayed successful status is insufficient. An expired, denied, malformed or unavailable Suite answer keeps new activity paused. Independent plans do not make this paid-license check.

Open and read License & updates

As a site administrator, open **Settings → Free Gifts & BOGO License**. You can also use **License** on the plugin's WordPress Plugins row or in the Free Gifts & BOGO workspace. This is the product's native license page, including when the rest of the workspace is inside Suite. It remains available when the normal WooCommerce workspace cannot finish loading.

Managing the store and managing site licenses are different permissions. A merchant who can use the Free Gifts & BOGO workspace may still need a site administrator to activate, release or reset its license. Administrators see a **Free Gifts & BOGO: warning** and **Open License** link when a licensing or policy condition needs attention.

Detail	How to read it
Release	The product's displayed release label.
License status	Current Free Gifts & BOGO product entitlement, such as Licensed or Activation required.
Promotion activity	Enabled, or Paused with saved data available. This includes any required Suite permission and server activity restriction.
Licensed site	Website address retained for this installation.
Environment	Its selected licensed environment.
Suite	Independent entitlement or the current required-Suite condition.
Last attempt	Last validation or remote-release attempt recorded locally. Initial activation does not update this field.
Last accepted check	When the service last accepted this installation, using the service's timestamp.
Next automatic check	Due retry or routine check. A due time is not proof that WordPress executed it.
Current permission deadline	The earlier applicable product term or connection deadline.
Updates	Separate protected-update readiness.

Times are shown in UTC to the minute. An em dash means no value is available for that display. Some recovery states show no computed deadline even when a historical term was retained. The current status and promotion-activity rows take precedence over a past success date.

The page does not display your complete key or saved installation secret. A licensing form result is shown once as an informational notice and expires after two minutes. Read the current status after the notice: **The licensing action completed. Review the current status below.** does not mean every separate policy condition is now satisfied.

Activate or replace the Free Gifts & BOGO license

1. Open **Settings → Free Gifts & BOGO License** as an administrator.
2. Find the complete Free Gifts & BOGO key in your AffStackDev account.
3. Under **Activate or replace your Free Gifts & BOGO license**, enter it in **Free Gifts & BOGO license key**. The field is masked, starts empty and has a browser limit of 128 characters; the client's limit is 128 bytes.
4. Select **Site environment**: Production, Staging, Development or Local. It initially follows WordPress's environment and becomes locked while a readable credential is saved.
5. Select **Activate / replace license** and review both **License status** and **Promotion activity**.

The key is used for that activation request. The plugin then retains an encrypted installation credential for later validation and updates. Secure local encryption support is required; if the host cannot encrypt the credential, correct that support before retrying. The product does not save the complete key as an update workaround.

If no activation slot is available, release an unused activation in the account or contact AffStackDev. The plugin does not remove another site's activation automatically. A server-side site-binding conflict must be resolved before reusing that installation.

A replacement first saves a protected candidate while retaining any older active credential. If the response is lost, the server may already have accepted the candidate. Select **Check now / recover pending activation** before submitting the key again. A readable pending candidate is explicitly refused by another activation request until it is resolved.

The activation button remains visible during recovery states, but visibility does not guarantee that the request will succeed. Read the status first. A changed identity or unreadable credential may need the explicit local reset described below. If a remote deactivation is pending, use Check now to finish that intent before attempting a replacement.

A rejected replacement does not automatically erase an older independent accepted credential. Continued use still depends on its deadlines and policies. The licensed environment cannot be changed while credential storage remains bound; release or resolve the current installation first.

Checks, grace periods and recovery states

A successful acceptance normally schedules another check about 12 hours later, spread by up to 15 minutes in either direction. Previously accepted product permission can survive a connection interruption for up to 72 hours from local acceptance, or until an earlier server-authorized term/grace deadline. A retry does not restart this period. At the deadline itself permission stops.

A server-authorized term grace period is distinct from connection grace. The product does not invent a new paid term after expiry. A future term start or an already ended returned term/grace is not accepted as current permission. If the account was renewed or its activation corrected, **Check now / recover pending activation** obtains a fresh answer using the retained installation credential.

Status label	Meaning and recovery
Activation required	Activate this product's own key.
Activation recovery pending	A saved candidate has no confirmed active credential. Use Check now; automatic recovery can also resolve it.
Validation required	A credential exists without an accepted usable state. Check now.
Licensed	Product entitlement is accepted within its deadline. Also review Promotion activity and Suite.
Licensed — validation overdue	Routine validation is late but the hard deadline has not passed. Check now if automatic recovery has not completed.
Licensed — connection grace	Latest check could not reach the service; restore connectivity before the retained deadline.
Licensed — term grace	The service granted a limited term grace period. Review the account before its end.
Licensed — term and connection grace	Both conditions apply; the earlier deadline wins. Resolve account and connection conditions.

Status label	Meaning and recovery
Licensed — deactivation pending	Remote release is not yet confirmed. Check now retries the same saved release intent.
License validation expired	Accepted deadline is absent or reached. Recover the license connection and entitlement.
Activation missing	The service did not find this exact active installation. Review its correct activation.
License could not be authenticated	The saved credential/product scope was rejected. Check the correct Free Gifts & BOGO product/account.
License unavailable	The service does not disclose whether the cause is expiry, suspension, revocation or another restriction. Review the account.
License needs attention	Authoritative denial needs account/activation recovery.
Local license state needs recovery	Cached accepted state failed its integrity check; the encrypted credential was retained. Check now.
System clock needs attention	Local time moved backward beyond the five-minute tolerance. Correct the clock, then check.
Site identity changed	The site's address or WordPress environment no longer matches. Resolve the move or use explicit reset.
License identity needs recovery	Retained accepted binding differs from current configuration. Check now to refresh it; this label alone does not necessarily show the reset form.
License credential needs recovery	Saved encrypted credential cannot be read. Restore the correct cryptographic configuration or use explicit local reset.
Site deactivated	Remote release was confirmed. Enter a valid key when this site should be activated again.

Check now / recover pending activation also retries a saved pending deactivation before ordinary validation. Its label remains the same. A form error may be more general than the stored status, so review both. Exact developer codes and precedence are in the technical error reference.

Automatic retries and safe visits

After the first local/connection failure, retry is roughly 15 minutes; after the second, roughly one hour; later retries are six hours. A numeric service throttle can raise a connection retry up to six hours. A server denial instead returns to the routine schedule. None of these attempts extends the already accepted permission deadline.

Working WordPress scheduled tasks provide the normal recovery path. A safe administrator page visit can retry an overdue check, and ordinary public browsing can queue a short check at the end of its request. On hosts that support it, the response is finished before that check; other hosts use the same four-second automatic network timeout during shutdown.

This browsing recovery excludes cart, checkout, account, order actions, API/AJAX requests, command-line work and other mutations. It never asks the shopper to enter a product key. AffStackDev products share one

automatic check budget on such a visit, so another product can defer Free Gifts & BOGO's attempt. A low-traffic site still needs scheduled tasks or an administrator's explicit check.

Automatic scheduling requires a readable credential. An unreadable encrypted envelope needs administrator recovery rather than repeated automatic checks. If an action is already in progress, wait briefly and retry. Activation, validation and remote release use a short lifecycle lock; do not start a local reset concurrently with one of those actions.

Move a site or recover an unreadable credential

The installation is bound to its site address and WordPress environment. Its encrypted credential also depends on the configured service, licensed environment and WordPress security salts. Moving or cloning a site, changing its address/environment, or replacing security salts can make the retained identity or credential unusable.

Changing the WordPress environment type (`WP_ENVIRONMENT_TYPE` in `wp-config.php`), for example when a staging copy becomes your live site, counts as a changed environment. Release this site's activation first ([below](#)) while the identity still matches; after the change, reset the local identity as described here and activate again, so the released place is free for the new identity.

When the screen offers **Moved or cloned site recovery**:

1. Decide which site should use the activation and review the original activation in the AffStackDev account.
2. If the problem came from an accidental change to the original site's security configuration, restore the appropriate configuration where possible.
3. Ensure no activation/check/release is still running.
4. Tick **Reset this site's local licensing identity and credential**.
5. Select **Reset local licensing identity**.
6. Review the new status and activate this site with the Free Gifts & BOGO key.

This creates a new local installation identity even when the problem was only an unreadable credential. It clears local accepted license state and credentials, preserves promotion settings and orders, and makes no remote release request. Arrange release of the original server activation separately; otherwise its slot can remain occupied.

The reset form appears only for a changed current-site identity or unreadable credential. A damaged accepted-state cache with a readable credential normally uses Check now. A healthy installation does not offer a general reset button. If local storage is failing, resolve that problem before relying on any completed-action notice or repeatedly creating new activations.

Release this site's remote activation

Use **Release this site's activation** when you want the server to free this site's Free Gifts & BOGO activation. Deactivating or removing the WordPress plugin is different and does not release that slot.

1. Resolve any pending activation with **Check now / recover pending activation**.
2. On the correctly bound site, tick **Release this site's Free Gifts & BOGO activation and pause its promotion activity**.
3. Select **Deactivate license on server**.

4. Confirm that status becomes **Site deactivated**.

The form is always present, but a successful release requires a readable active credential and matching current site identity. With no active credential it reports failure. If the request is interrupted, the token and original release request identifier are retained. Check now or automatic recovery retries that same request; do not assume the slot was released before acknowledgement.

Product permission may remain usable within its prior deadlines while release is pending, subject to activity policy. Confirmed release clears the local credentials, pending intent and scheduled validation while retaining configuration and business records. You can later activate again with a valid key. A moved copy cannot use this control to silently change the old binding; manage the original activation separately.

Protected plugin updates

Free Gifts & BOGO checks its own protected release service, verifies the publisher's signature and confirms package contents before accepting a protected update. It needs an accepted product entitlement, readable installation credential, service support for installation-token updates, a trusted public release key, PHP Sodium and ZipArchive. Your store does not need or receive the publisher's private signing key.

Updates label	Meaning and next step
Waiting for token-authenticated updates	Current accepted response does not advertise the required update capability. Recover/validate the product license and use Check now after service support is available. The complete product key is not retained as a workaround.
Release trust key required	Service capability exists but local public-key or verification extensions need attention. Restore the authentic package/key and ask the host to enable missing extensions.
Free Gifts & BOGO license required	Verification support is available but current product permission or readable active credential is missing. Recover licensing.
Protected updates ready	WordPress can check and download authorized signed releases. This does not opt the plugin into automatic installation.

Readiness checks service capability and trust support before displaying the missing-license label, so a new unactivated installation can first show the token-authentication message. These update labels are separate from **Promotion activity**; a plan's Suite hold or server activity restriction alone does not necessarily prevent a product update.

An administrator who can update plugins sees **Check for plugin updates**. A successful fresh check refreshes WordPress's plugin update information. **Last update check**: shows its UTC time and Up to date, Update available or Update check needs attention. This is separate from the license attempt/acceptance timestamps and is not an installation history.

Enable or disable automatic updates in **WordPress → Plugins**. Protected manual and automatic updates use the same verification path. The plugin offers newer stable releases and checks the release's PHP, WordPress and WooCommerce requirements. It does not force automatic installation.

Routine metadata and failed-check cooldowns last 15 minutes. The explicit button requests a fresh check. Download and prepared installation require fresh permission and the same current artifact, so an old update notice is not permanent authorization.

Recover a stopped update

Problem	Recovery
Service or transport unavailable	Restore connectivity/TLS, then check again. A throttled service may need time before retry.
License/credential/identity unavailable	Recover the product's license or complete explicit site reset/activation where appropriate.
Host below release requirements	Meet the signed PHP, WordPress or WooCommerce minimum before retrying.
Temporary ZIP cannot be created	Have the host correct temporary-file storage and permissions.
Signature, ZIP or extracted-file verification fails	Obtain intact publisher-signed bytes; do not edit proofs or force through a failed protected check.
Signed installed file changed	Review custom edits, missing files and invalid proof files; reconcile them with the authentic signed release.
Offer or authorization changed	Refresh license/update information and retry the current offer.

The installed-file check verifies the files listed in the signed manifest and their containment; it does not promise to detect every unrelated extra file in the plugin directory. The downloaded signed ZIP and its prepared extracted source must match their exact inventories. An installation with neither proof file can receive its first signed update; one missing proof is an error, not an unsigned-installation exception.

The update marker is not a reusable download credential. Temporary grant authorization is obtained for downloading, and tracked temporary files are cleaned up locally. An update HTTP rejection is reported as an update error; this client does not automatically convert it into a product-license denial. Review the actual license screen when diagnosing it.

The protected verification sequence applies to updates prepared through this product's protected download path. It does not claim to authenticate every manual filesystem replacement. Preserve the exact error and consult the technical error lookup when a package/service problem persists.

Work inside Suite or use the native pages

When a compatible Suite is already active, Free Gifts & BOGO registers its product workspace and **Free Gifts & BOGO summary** dashboard contribution. It does not install Suite or take ownership of Suite's menus, layout, shared storage or licensing. The normal workspace still requires WooCommerce and your store-management permission.

Suite can show Dashboard, Reports, Coupons, Send a gift, Preview, Settings and Promotions in place. Product navigation also provides ordinary PHP destinations, including a dedicated Suite product page where available. The **License** destination always opens the native license page. If Suite becomes unavailable during an in-place request, reload and use the standalone Free Gifts & BOGO workspace or the independent license page as appropriate.

Enhanced presentation needs both the product's saved Enhanced choice and Suite's parent permission. A Performance preview or a disabling integration filter can reduce it further; it cannot enable a layer that either

owner disabled. Suite's product registration reports the local preference, while the rendered section reports its current effective display mode.

In-place actions retain each native operation's security checks. Promotion actions use their record/revision checks; coupon actions require the current operation/record confirmation token; settings, policy-page creation and preview use their own form checks as well as the Suite request token. A held promotion license allows loading and read-only preview, but pauses mutation actions.

If an action or coupon link has expired, refresh that section and try again. If a message says **The action may have completed, but Free Gifts & BOGO could not refresh the result. Reload this page before trying again.**, reload and inspect the current saved result before repeating the action. A failed refresh is not proof that a preceding save was rolled back.

Read both the refreshed result and its message. Some coupon or policy errors are shown inside a successfully refreshed section, while a licensing, unavailable-section or expired-token failure can reject the request itself. Search, paging and promotion history links are navigation state; they do not grant permission to perform an operation.

Retained license data and uninstall choices

License requests send this product's installation/site/environment details to its configured licensing service. Activation also sends the full key once and a proposed installation credential. Later validation/release and protected metadata requests use the installation credential; the final archive download uses a short-lived grant credential. Stored installation credentials are encrypted using site-specific configuration and WordPress security salts.

Normal plugin deactivation stops scheduled license checks and retains licensing and business data. Default uninstall also retains data. Neither contacts the service to release the remote activation. Confirmed **Deactivate license on server** releases that activation and clears local credentials while keeping saved promotions and orders. **Reset local licensing identity** creates a new local identity and clears its credentials, but leaves the old remote activation to be released separately.

If you explicitly enabled deletion on uninstall, removing the plugin also deletes its local license configuration, state, credentials, pending operation and lock, its update records, and the promotion/settings/cache data described in the uninstall section. WooCommerce history and retained campaign claims remain. This opt-in deletion still does not release a remote license slot. Release the activation through the supported license control before removal when that is your intention.

Do not manually edit encrypted values to recover a site. Restore the appropriate configuration or use the supported recovery controls, then confirm the current status. A copied credential is not a portable product key, and an old license success date does not override the current installation's identity or deadline.

Free Gifts & BOGO's Commerce Core connection

Free Gifts & BOGO includes Commerce Core release 0.1.0.2c, native version 0.1.0.2.3. It offers that copy when WordPress loads the plugin. If compatible AffStackDev products offer other copies, the loader selects the highest supported version; the product named as host supplies the runtime code. Hosting does not give that product control of Free Gifts & BOGO campaigns or gifts.

Core coordination requires 64-bit PHP 8.1 or later, WordPress 6.7 or later, and WooCommerce 10.0 or later. Free Gifts & BOGO's own package supports PHP 7.4. Native, unconnected promotions can run when its valid Core offer is refused solely because the platform cannot run Core, provided no previously required Core/B2B connection remains and the product's own permission and promotion conditions pass. Corrupt runtime files, conflicting copies or a missing required provider do not qualify for that fallback.

Free Gifts & BOGO records required connections after using them. Removing a companion or lowering the PHP version does not erase that requirement. If a previously connected store becomes held, restore the supported runtime or follow the outgoing product's supported retirement workflow. Free Gifts & BOGO does not provide a switch that forgets these connections.

Read Integration status

Open **Promotion Settings** and find **Integration status** below the settings form. A user with the product's management capability, `manage_woocommerce`, can read the section. It works with JavaScript disabled and remains useful when licensing pauses new promotions.

Row or part of the message	How to use it
B2B / Commerce Core	Explains whether B2B registration is confirmed, Core is available, the narrow standalone path applies, or a connection is held. Follow a held message to B2B Tools or restore the supported runtime.
Core runtime; host; version	Reports the selected copy; none means that value is unavailable. A loaded runtime is one prerequisite, not confirmation that a particular shopper can receive a gift.
Audience	Summarizes the current account's returned audience status/reason. The promotion engine performs its full fresh checks separately.
B2B retirement	accepted means the supported absent-owner proof can be verified; not_verified is not an accepted removal.
Promotion permission	Explains the product's license permission. Customer eligibility and runtime requirements remain separate.
Campaign allowance storage	Reports whether limited campaigns can use their checkout allowance table. A database problem here must be repaired before a limited campaign can award gifts.

B2B integration requires B2B 0.8.1 or later and Commerce Core 0.1.0 or later, with contract 1 and their required capabilities. Free Gifts & BOGO checks the actual installed versions and declarations. A version label without the required functions/capabilities is not sufficient. Compatible updates can be installed independently when these contracts remain supported.

Free Gifts & BOGO owns the gifts. Its status states that no equivalent successor is registered. This is not a claim that another product's shared adapters are absent; it means Free Gifts & BOGO does not hand its campaigns to a replacement executor. This section has no Core version selector, gift ownership-transfer button, retirement form or general shared-feature editor.

How current eligibility controls gifts

Free Gifts & BOGO checks current product permission and customer eligibility before promotion effects. Confirmed B2B customers and consumers excluded by the business policy do not receive consumer gifts; they see no Free Gifts box in the cart or at checkout. Unknown, stale or provisional audience information also prevents a new gift; Free Gifts & BOGO does not turn an unavailable business-policy connection into an eligible customer.

On an ordinary store that has no audience provider, Core can report that no audience policy applies. Free Gifts & BOGO can use its own campaign conditions in that situation. This is different from a previously required B2B provider disappearing: the latter keeps a hold until the provider is restored or its supported retirement is accepted.

Current eligible evidence is short-lived and is checked again before changes. An earlier cart display, a status message, an eligible B2B callback, or a prior successful request is not a saved authorization for the next change. If the customer, policy revision or relevant cart context changes during an operation, the request is held and the cart must be reviewed again. Renewal of the same evidence's expiry alone does not change the business decision.

When B2B has been deliberately retired

Use the outgoing product's supported retirement controls while that product is still available, then complete acceptance after deactivation as directed by its interface. Free Gifts & BOGO itself only reads the result. Its accepted proof must identify B2B, the current site and the absence of live audience policy; the runtime's retained policy state must still match.

With that verified proof and B2B absent, Free Gifts & BOGO can remove the missing-B2B policy hold for current native promotions. It still applies the product license, campaign conditions and gift-line safety checks. Retirement does not rewrite old orders or declare historical customers eligible, transfer business pricing, or remove stored provider expectations. Returning B2B or changing the bound runtime policy causes the removal proof to be checked again.

Protected gifts and cart recovery

Free Gifts & BOGO removes or modifies only a positively identified gift it can safely control. It checks its reward marker and any active Core or Commerce Bridge line protection. Protected, unrecognized or interconnected lines are held for review. A gift with another owner's provenance, commercial facts or relationships is not silently rewritten as an ordinary Free Gifts & BOGO line.

When B2B or current permission withdraws eligibility, Free Gifts & BOGO attempts to remove its verified gifts. A malformed tag or uncertain ownership can stop cleanup; any gifts already removed stay removed. Paid items and unrelated items are not removed as a shortcut. The checkout message is **A free gift could not be verified safely. Please refresh your cart and review your gifts before checkout.**

1. Refresh the cart and review its current gifts and totals.
2. If the message remains, check **Promotion Settings → Integration status** for a license, B2B, Core or storage hold.
3. Restore any previously required companion and its supported integration. For intentional B2B removal, complete the supported retirement procedure.

- 4. Review a protection or ownership conflict through the product that owns that line; do not remove internal metadata to force checkout.
- 5. Continue only after the refreshed cart is accepted. Cleanup during a failed checkout can still require another review before payment.

The system retains the prior Commerce Bridge protection requirement when it has been observed. Removing Bridge does not convert a protected or uncertain gift into an ordinary removable item, and B2B audience retirement does not remove that separate protection requirement.

Free gift offers in a customer-referral program

When a compatible Store Credit customer-referral program asks about other promotions, Free Gifts & BOGO supplies read-only facts about the gifts actually present. The referral program decides whether those offers may combine with its benefit. Free Gifts & BOGO does not create a referral program, issue credit or change a referral coupon through this connection.

The companion's named selector can distinguish:

Selector	Meaning
Spend-based free gifts / Quantity-based free gifts	The qualification type of the awarded gift's campaign.
Single reward / Tiered reward	The campaign's reward structure.
Automatic gift / Customer-choice gift / Same-product BOGO	The awarded reward method. A tiered campaign is classified by the awarded tier's marker, not an unrelated hidden root method.
A named campaign	A specific Free Gifts & BOGO rule, labeled with its qualification type.

The selector always includes the seven general choices, plus up to 40 matching published or draft campaigns. Search is limited to 80 bytes; the list has no page control in this adapter. A draft appearing in a selector does not activate it. The actual gift still needs meaningful validated campaign evidence.

In a live cart, Free Gifts & BOGO recognizes a correctly marked gift whose product matches its marker, quantity is exactly one and price is zero. If line subtotal/total have already been calculated, they must also be zero. A corrupt marker, nonzero amount, changed product, invalid campaign or unresolved evidence returns an unknown offer; it does not quietly report that no gift exists.

Disabling a campaign does not erase the description of a gift that has already been awarded in the cart. Its configuration must still be valid. The referral program receives the current cart fact and applies its own combination policy. It can handle unknown evidence conservatively without changing Free Gifts & BOGO's campaign configuration.

Referral compatibility on existing orders

New gift order lines can retain a separate frozen description of their campaign ID, qualification type, reward structure and reward method. That record is bound to the line's normalized reward attribution, including its retained names. It is saved at creation, hidden from the ordinary item-metadata display, and used when a companion later evaluates historical offer compatibility.

Old attribution by itself does not prove the campaign's historical type or reward structure. Free Gifts & BOGO does not fill a missing frozen record from today's campaign or backfill an existing order when a checkout hook runs again. A renamed, edited or deleted current campaign therefore cannot supply missing historical proof. Changing a saved line's attribution can also make its frozen binding invalid.

A readable historical offer still requires the product/variation to match the retained attribution, quantity exactly one, and zero subtotal/total. If a snapshot is missing, altered, unsupported or inconsistent, the referral adapter reports unknown. The companion program owns the resulting eligibility decision and any existing credit or referral reward; Free Gifts & BOGO does not reconstruct, pay or reverse those obligations.

Resolve integration messages

Message or situation	Next step
B2B did not accept Free Gifts & BOGO registration / integration	Open B2B Tools, resolve its active integration hold and reload. Both products must supply the required contract and capabilities.
Free Gifts & BOGO could not retain its Commerce Core connection requirement	Ask the site maintainer to repair WordPress option storage. Preserve the recorded connection requirement.
Unrecognized B2B compatibility descriptor	Restore a consistent complete B2B release; the declared identity must match the installed product.
Bundled Commerce Core unavailable	Check the selected host and the Core platform requirements. A corrupt/conflicting copy requires repair, not standalone fallback.
Companion check could not finish / required support missing	Resolve the named companion's diagnostics or install a compatible release with the required interface.
Gift could not be verified safely	Refresh and review the cart, then follow the ownership/protection and eligibility recovery steps above.
Historical referral offer is unknown	Preserve the old order. Use the referral program's supported review policy; today's campaign settings cannot prove missing historical facts.

Connection messages explain availability; they do not replace individual gift eligibility checks. The technical reference contains the exact diagnostic codes and field contracts for an integration maintainer.

What Commerce Core does

Commerce Core connects compatible products to shared customer context, cart-line identities and accepted order records. Free Gifts & BOGO includes the runtime and offers it for selection when its requirements are met. It needs 64-bit PHP 8.1 or newer, WordPress 6.7 or newer, and active WooCommerce 10.0 or newer. The included Core release is 0.1.0.2c; its technical version is 0.1.0.2.3. See [Free Gifts & BOGO's Core connection status](#) for the product's own Integration status display.

When several compatible products include Core, the highest eligible runtime version supplies the code for that request. Matching versions must contain matching listed runtime files; conflicting copies are held until the affected products are updated to a consistent release. An already loaded standalone Core is not loaded a second time. The **B2B / Commerce Core** message in Integration status identifies the selected runtime status, host and

version. The selected host is not automatically the owner of your prices, gifts, business policies or credit balances.

Core coordinates information and records. Your connected products still control their business features and licensing. Installing Core alone does not create a discount, assign a customer to a business program, issue credit, transfer a gift rule or add a new payment action. Features that need a product adapter stay unavailable until that compatible adapter is present.

Read shared records and technical details

Free Gifts & BOGO's Promotion Settings screen displays its integration status, as described [above](#). The bundled Core also contains a detailed panel that a compatible host can expose. When AffStackDev Suite supplies that interface, open **AffStackDev → Suite Settings → Commerce Core**, then expand **Shared records and technical details**. Viewing that host panel and downloading its records requires permission to manage WooCommerce and access to the host. The forms work without JavaScript. Free Gifts & BOGO does not independently add this panel or register its download action.

Display	How to read it
Commerce Core version	The version of the runtime selected for this request. A compatible connected product may supply a newer runtime than Free Gifts & BOGO's included copy.
Foundation — Available	Core can read its local runtime configuration. Connected products still have their own readiness checks.
Foundation — Needs review	Core cannot read the required local record. Resolve the database or stored-record problem before relying on affected shared features.
Audience connection — Configured	Core remembers at least one product responsible for determining consumer-promotion eligibility. This is a retained requirement, not proof that the product is currently working.
Audience connection — Not configured	No audience owner has been configured in Core's retained record. Each product continues to apply its own ordinary consumer behavior and other conditions.
Order records	Count of accepted Core order snapshots. If the database count fails, the card can display 0; the downloaded status report distinguishes that failure from an actual empty history.
Audience providers	Currently registered owner names, their declared versions and registration state. This list does not itself run their eligibility checks.
Shared feature owners	The last saved feature assignment, with its feature ID, capability, owner and current recorded state. Products verify permission and availability again before applying a change.

The shared-feature state reads **Operation needs reconciliation or is in progress** when an operation is pending, **Enabled** for an enabled unarchived definition, or **Paused** otherwise. **No assigned owner** means no owner is recorded; **A shared record needs review** means the returned entry cannot be presented normally. If no shared adapters are available, the panel explains that none are registered for this request. The list displays the first 25 shared records. An integration can obtain additional records through the paginated status interface; there is no Next button in this panel. The [shared-feature chapters](#) explain ownership, editing and operations.

Respond to connection messages

Connections need attention can show more than one issue. Resolve each applicable condition rather than treating the Foundation card as approval for every feature.

Message concerns	What to do
Local records unavailable	Restore database access and readable Core storage. Reload the panel after repair. Existing records remain the basis for recovery.
B2B needs its Commerce Core reader update	Use a B2B integration that supplies the supported fresh reader. The older reader can retain business exclusions but cannot approve uncertain consumer promotions.
Dynamic Pricing has multiple audience callbacks	Have the owning integrations remove the competing adapter as part of their coordinated update. One audience decision path is required.
More than one expected audience owner	Review the competing integrations and their retained requirements. Core does not choose an owner by loading order or silently discard one.
Previously configured audience provider missing	Restore that product's supported integration or follow the explicitly supported policy-retirement procedure for the affected consumer. Simply deactivating a provider does not erase Core's expectation.
Previously configured licensing protection unavailable	Restore the expected protection integration. A missing reader does not turn protected or unknown merchandise into ordinary merchandise.

A compatible Suite host can also provide its own product connection report. Accepted retirement is a separate proof that integrations may explicitly consume. This bundled Core's ordinary audience and line-protection readers do not automatically clear their remembered requirements when a retirement proof is accepted.

Download reports and individual order records

When a compatible host exposes the Core panel described above, open its **Records and support** section:

1. Select **Download status report** to download `commerce-core-report.json` with current runtime, provider, connection, snapshot-count and shared-feature information, plus the latest retained diagnostic sample.
2. To investigate a specific order, enter its positive numeric **Order ID** and select **Download order record**. The server accepts an existing order ID of 1–13 digits, without a sign, decimal point, spaces or leading zero.
3. Keep the file with the relevant support or integration investigation. Downloading changes neither the order nor its journal state.

Both downloads use the same filename. Rename saved copies if you need to retain more than one report. An individual order download contains the journal's revision, state, fingerprint and recorded data. It can include a draft or an interrupted publication as well as a committed record. It does not convert an unfinished record into an accepted one.

Enter a valid order ID means the input format or order lookup failed. Correct the ID and confirm the order exists. **This order has no readable Core record** means there is no readable Core journal snapshot for that order. A real WooCommerce order can exist without a Core snapshot, for example when no compatible historical-audience provider was configured. The export does not substitute today's customer classification or an older B2B record. **Access denied** requires a user with WooCommerce-management permission. If a form's security verification expires, reload the Settings screen and use its fresh form.

Reports are administrative records. Status reports identify integrations and saved shared definitions; order reports contain the recorded order subject and line evidence. Provide the relevant file to the person authorized to investigate the issue.

Customer eligibility and pricing integrations

An audience owner determines whether the current shopper is eligible for consumer promotions. Core ties its decision to the store, shopper and current checkout context. Changes in login, cart contents or quantities invalidate the request's previous context. Currency, shipping choice, tax scope, variation attributes and applied coupons also contribute to that context. Core passes a digest rather than storing the shopper's address in these context fields.

Fresh decisions have a limited validity period of at most five minutes and must match the same request. Consumer eligibility requires a consumer classification. Invalid, missing, conflicting or expired evidence leaves eligibility unresolved; it is not a permission to apply a consumer promotion. Connected products may also have their own finality and business-rule requirements.

The **Configured** state is deliberately retained when an audience provider disappears. Removing the plugin does not make all shoppers consumers. If no owner has ever been configured, Core reports that no audience provider applies; the connected product decides how its own standalone behavior uses that result. Core does not expose a customer selector, manual audience override or force-eligible button.

Legacy B2B and Dynamic Pricing

The older B2B reader can preserve a known ineligible outcome. It cannot establish fresh eligibility for a consumer, even when its earlier result said eligible. Its historical reader also cannot prove that an order's business context was atomically finalized. Install the supported reader integration when the affected product requires fresh or accepted historical evidence.

Core's Dynamic Pricing adapter runs only when an audience dependency already exists, B2B is present, or the runtime configuration cannot be verified. It requires exactly one audience callback. It delivers fresh supported owner evidence to Dynamic Pricing; it does not calculate the discount itself. Competing callbacks or uncertain B2B evidence hold the affected consumer decision until the integration is corrected. Installing Core by itself does not turn a previously independent Dynamic Pricing setup into a new audience-dependent setup.

Commercial facts without price changes

A compatible product can attach a commercial fact to a cart line, including currency, quantity, tax basis, price source, lock state, an optional unit amount and the kinds of adjustments its contract recognizes. Refreshing that fact replaces the prior fact in the same owner/slot. Core records the information; the owning product must actually apply any permitted price or adjustment.

Stored facts can remain on an order after their current-request validity expires. They describe what was recorded and do not approve a later price change. Missing amounts are distinct from zero. For integration authors, value rules and commercial-fact fields give the exact decimal, field and size requirements.

Shared cart lines and protected merchandise

Compatible products can opt a cart line into Core identity. An untouched ordinary line does not receive an identity merely because Core reads it. Once a product claims or links a line, Core preserves its UUID, the

declaring product's reference and any declared relationships or facts. This allows connected products to describe a line consistently without changing the product itself.

A line can contain up to 16 provenance entries, 32 relationship entries and 16 commercial facts; Core cart operations support at most 200 lines. A product updating its own owner/slot replaces that entry rather than adding another identity. A relationship creates matching outgoing and incoming records on two different lines. Removing that particular relationship keeps the lines' identities and other records. These are integration operations, with no general claim/link/unlink buttons in Free Gifts & BOGO's Promotion Settings.

If Commerce Bridge supplies the existing line identity, Core reads it as Bridge-owned and does not migrate it. While the Bridge cart reader is present, Core's own line writers stay read-only, including for a line that Bridge describes as unprotected. Core also refuses an unknown protection result. An integration must establish one supported writer rather than let two systems alter the same identity graph.

Core reports relationships pointing to a cart line that is no longer present; it does not silently remove them. Complete order snapshot creation requires every related UUID to exist in the order. Duplicate IDs, mixed Core/Bridge identities, malformed records or unresolved protection need correction by the owning integration. If checkout displays **A shared cart line needs review. Please refresh the cart before checkout.**, refresh and review the cart. If the same message returns, investigate the line-owning and protection integrations before repeating checkout.

Core's classic-checkout line hook stores the validated line record in WooCommerce order-item metadata. Blocks snapshot capture reads whichever Core line records actually exist on its order items; the bundle does not register a separate Blocks line-record writer. Normal formatted order-item metadata and normal WooCommerce REST order responses omit raw `_afsd_cc_` records. Authorized Core readers and exports remain the way to inspect their recorded evidence.

Accepted order history and publication recovery

An accepted Core snapshot preserves the owning provider's committed audience decision together with the recorded order and line context. It includes the recorded customer, currency, totals, payment-method identifier and line evidence. The order fingerprint binds those fields to a specific order state. A snapshot does not itself mean the order was paid or authorize a commission, reward or credit.

Automatic capture is available on classic and Blocks checkout only when exactly one expected audience owner has a registered historical-order reader. A current eligibility reader alone is insufficient. If no such owner is available, Core does not manufacture a snapshot. This means the Order records count need not match your total WooCommerce order count.

Once accepted, history remains readable even if the customer changes, the order is refunded, the provider is removed or the order's current totals change. Those later events do not rewrite the original snapshot. Its purpose is retained historical context, not a fresh entitlement for today's shopper. An older B2B record can be returned only as unverified legacy history; it does not become a committed Core record.

Draft, publishing and committed

State	Meaning and next action
Draft	Core has prepared matching order and owner evidence but has not finalized publication. A supported integration can continue using the current revision and original order fingerprint.

State	Meaning and next action
Publishing	Core saved the durable publication intent before confirming the corresponding order metadata. Preserve that intent and investigate the original record.
Committed	The exact order projection was confirmed and the journal finalized. The accepted record remains available for historical reads.

If the panel reports **An interrupted order publication needs review**, download the original order record. A user with WooCommerce-management permission can have the supporting integration invoke Core's reconciliation API with that exact order, revision and fingerprint. There is no reconciliation button in this panel. Reconciliation confirms that the intended order projection already exists and then finalizes the journal; it does not write a second projection or replace the original request.

If that projection is missing or differs, the operation remains pending. Passage of time does not authorize a takeover or discard the intent. Resolve the saved-record discrepancy through the owning integration. Do not create another order or use a different operation identity as a substitute for confirming the original result.

At checkout, **The order context needs review before payment** means Core could not verify the order fingerprint. **The accepted order context could not be saved. Please contact the store before retrying payment** means preparation or publication failed. Inspect the existing order and its retained Core state before retrying payment. A late failure can leave a publishing record that requires reconciliation rather than a fresh attempt. Technical procedures are in snapshot publication and snapshot recovery.

Diagnostic reports and integration interfaces

The status report includes diagnostic counters from the current request and the latest retained affected request. Core retains up to 32 distinct bounded code names per request, with each counter capped at 999. It saves one latest sample rather than an accumulating activity log. An ordinary “no audience provider” outcome is not stored as a persistent warning. A later request with no new relevant codes leaves the earlier sample intact, so compare its recorded time with the current problem.

Diagnostic samples contain code names, counts, Core version and recording time rather than raw provider error text or shopper details. The broader status and order reports contain their respective integration and order records. A report code identifies the failed boundary; it does not mean every previous step was rolled back. Runtime error recovery and [shared-operation recovery](#) explain how to interpret the retained state.

Developers use Core's local PHP interfaces to register readers, read current audiences, manage line declarations, prepare and reconcile order snapshots, and call the separate shared-definition, operation, retirement and checkout-participant protocols. There is no public HTTP audience endpoint. Most local interfaces rely on the calling integration to verify its own user permissions and request security. Core adds WooCommerce-management checks to its administrative downloads and specified management/reconciliation operations. The complete function reference identifies every facade, argument order and owning contract.

Core strings use the `affstackdev-commerce-core` translation domain. This bundle does not contain a translation catalog or register a separate catalog-loading hook. Core's status layout wraps long identifiers, stacks cards and download fields on smaller screens, and provides visible keyboard focus for its controls. Its included GPL text is available with the runtime files.

Retained commerce records and safe updates

Commerce Core keeps the records that let compatible products agree on a feature's current configuration, responsible product and unfinished operation. It does not become the owner of product prices, customer balances or payout ledgers. Orders continue to use WooCommerce's normal storage interfaces.

Free Gifts & BOGO includes Core 0.1.0.2c. Another compatible active host can provide the selected shared runtime. The available operation depends on that selected runtime and the owning products' supported interfaces, rather than on a release number alone.

Records belong to the original WordPress site. Core requires a supported 64-bit PHP environment and rejects attempts to reuse its active runtime after switching to another site. Its installer creates missing local records without rewriting an existing database schema. Network-wide installation through the Core installer is refused; activation must be handled separately for each site. An unavailable or unfamiliar record is preserved and held for recovery.

Each record has an identity, a revision, a state, its data and a fingerprint. A successful update must name the revision it actually read. If someone else saves first, the stale update fails instead of overwriting their work. A new record uses expected revision zero. For example, if two operators read revision 7 and one saves revision 8, the second must reload revision 8 before making a reviewed change. Saving at revision 7 again does not overwrite revision 8.

A fingerprint confirms the expected data content; it is not a payment receipt. Core can coordinate its records without guaranteeing that a separate financial service completed a payment. If a result is uncertain, keep the original reference and reconcile the owning service before taking another value-changing action.

Core also remembers which audience and line-protection services it has already depended on. Adding a requirement is persistent. Removing the product does not erase that requirement or make a previously protected benefit safe. There can be at most 16 expected audience owners. Repeated conflicting configuration writes eventually return a busy result; they do not silently drop an owner. Snapshot status counts report retained records by state, not the amount of money owed or paid.

These records have bounded, validated data. The journal accepts at most 128 KiB of canonical data per record; individual text values are at most 2,048 bytes, lists/maps at most 400 entries per level and nesting at most 12 levels. The responsible integration supplies supported fields and types. Merchants should use its current form or supported service instead of editing database rows or changing revision numbers manually. Exact storage contracts are in the technical journal reference.

Choosing a supported feature provider

A shared feature can run only through a product that explicitly supports its full configuration and is currently ready to perform the requested task. Products register their local adapters during normal initialization. This happens automatically through code supplied by those products; there is no screen for entering a PHP callback.

Provider registration identifies the owner, provider version, supported feature families and schema versions, preference priority, and separate availability, evaluation, execution and reconciliation functions. Up to 16 owners can register; an owner describes up to 16 feature families with up to eight schema versions each. These limits are integration boundaries, not a reason to divide a financial operation across several invented owners.

Core distinguishes checking or editing a feature, executing its effect and reconciling an earlier effect. A product may remain able to read an old receipt while it cannot perform new business. Both full feature support and

current task readiness must succeed. Missing callbacks, exceptions, a wrong site or conflicting registration keep the feature unavailable. A duplicate accepted owner can make the shared registry unavailable for the whole request; resolve the conflicting integration and begin a fresh request.

The ready preferred owner wins. If the preferred product is unavailable, Core considers other products that explicitly support the same definition. Among those, a smaller provider priority wins, with owner-name order breaking ties. For example, priority 10 precedes 200, but a ready preferred owner takes precedence over both. This is a technical adapter setting; it is not a ranking of merchant product discounts. A feature's own stored priority is separate from provider priority.

The provider overview reports registered or unavailable and its declared feature support. Registered does not mean licensed for this transaction, eligible to serve this customer or safe to issue money. Those checks remain with the product.

An authorized integration can retire a shared executor, or restore its availability, using the exact saved retirement-record revision and a reason. This requires WooCommerce management permission. Retirement prevents that owner from beginning a new authorized effect; it does not delete shared definitions, undo earlier effects or remove its ability to supply supported old-receipt reconciliation. A compatible alternative may become responsible after current authority is resolved. A pending operation is not forcibly transferred by this action. This executor control is separate from the [business-policy retirement procedure](#).

The technical provider contract defines the trusted local integration fields. It does not add a general provider editor to Free Gifts & BOGO.

Shared configuration, pauses and ownership

Shared configuration keeps one original feature identity while compatible products can edit or perform that feature under its supported contract. A product must understand the complete saved terms. Installing another product does not convert old terms or turn on a paused feature.

Configuration or record	Effect and restriction
Feature identity	The original unique ID is retained across edits, ownership changes and recovery.
Capability and semantics	Identify the feature family and exact meaning of its terms; existing records cannot change these identities.
Definition schema and payload	The owning product's complete supported field set. A replacement schema must be understood explicitly. Core does not infer missing settings.
Enabled	Allows execution when true and every other check succeeds. Turning it off retains the definition.
Archived	Holds execution while preserving configuration and history. It does not cancel a customer's accepted obligation.
Feature priority	Integer 0-9999 retained for the feature's own meaning; not the provider selection priority.
Preferred owner	Empty for normal supported selection, or the exact preferred supported product.
Creator and last editor	Preserve the original creator and identify the most recent changed definition.

Configuration or record	Effect and restriction
Definition revision	Advances when the complete configuration changes.
Ownership generation	Advances when configuration or selected ownership changes.
Record revision	Used to protect every saved replacement from a stale request. It is separate from definition revision and generation.
Pending operation	Holds the original operation reference, attempt token, expiry and evidence; blocks another feature operation or edit until resolved.

These fields are supplied through an owning product's form or an authorized local integration. Free Gifts & BOGO does not provide an unrestricted editor for arbitrary feature payloads. Configuration data must fit 48 KiB and the common text/depth limits.

An editor must currently support editing that exact definition. Saving a changed definition requires its exact record revision and no pending operation. It preserves the creator, advances definition revision and generation, clears the old owner and records the last editor. A save with identical terms leaves the row unchanged, but still checks edit authority and current revision. There is no shared-feature delete operation in this interface; disabling or archiving preserves the record.

When a compatible product asks who should act, Core checks whether the feature is enabled, unarchived and free of an unfinished operation, then selects a ready supported owner. If none is available, the saved feature remains held, potentially with an empty current owner. A missing owner never authorizes another product to guess the terms. Simultaneous ownership changes may return busy after three conflicting attempts; reload the current result.

The shared status view is read-only and requires WooCommerce management permission. It shows owner, generation, definition revision, enabled/archived state and whether work is pending. An unreadable feature is identified as unreadable rather than hidden as successful. Reporting is paginated: 25 records by default and at most 100 per request. A displayed owner is a stored assignment, not permission to execute now. For exact output fields see shared definitions and reporting.

Fresh evaluation and short-lived permission

A product evaluates a shared feature using its current configuration, current owner and exact transaction context. Core does not retain an old cart result as permission to apply it later. The selected product supplies the result, and Core checks authority again after that callback completes.

The returned authority claim lasts at most 60 seconds. It binds the site, feature, owner, ownership generation, definition revision and complete definition/context fingerprints. A changed customer or transaction context, changed owner, edited definition or expired claim invalidates the earlier evaluation. Obtain a fresh evaluation instead of editing or extending the old claim.

An unfinished operation has its own reserved authority and prevents ordinary resolution from supplying a competing owner. A local claim is not a public authentication token, product license, customer eligibility decision or financial receipt. The relevant product must still enforce each of those requirements.

Evaluation results are bounded to 16 KiB; context is at most 8 KiB. The owning product controls the supported fields. Integrators can use the evaluation and claim reference to preserve the complete envelope and correct data types.

One business operation, one retained identity

Every delegated operation uses a stable business reference and exact context. Core derives a stable operation and effect identity from that original reference and site. Keep it when retrying, moving between supported owners or investigating a failure. Creating a new reference to get around a pending result can create a separate effect.

Operation state	Meaning and next step
Pending	An attempt was reserved and its outcome is not yet confirmed. Reconcile the original operation.
Committed	The owner supplied a confirmed effect receipt. Repeating the original request returns the retained result without executing again.
Not applied	The owner supplied a confirmed no-effect receipt. Repeating the original operation returns that result; this bundled runtime does not reopen it for another effect.
Unknown callback answer	The journal remains pending; unknown is not a fourth terminal state or proof that nothing happened.

Core reserves the feature before asking a product to perform its effect. The feature permits one pending operation at a time. The operation freezes the feature definition, context, owner and short-lived claim, with a unique attempt token and empty receipt. Other operations and edits remain held while the reservation is unfinished.

Immediately at its financial action boundary, the product must confirm that the original pending record, token, lease, owner, generation and definition revision still match. Core reads this retained authority and then checks callback-based execution support and readiness. This implementation does not reread the journal after those callbacks. Retiring the owner or losing its current availability can stop a stale attempt. The product still owns its transaction, money lock, durable deduplication and authoritative receipt; a Core check cannot replace those controls.

A product returns a confirmed applied or no-effect result with its receipt. Core saves that terminal result before releasing the feature reservation. If reservation release fails, replaying the terminal original result can retry that release safely. A thrown callback, malformed answer or unknown outcome preserves pending evidence. A storage error can happen after the owning product applied an effect, so do not equate an error with “nothing was paid.”

Passive lookup retrieves the original record without executing anything. It is useful for support and reconciliation, but its existence alone does not make a pending effect safe to repeat. The technical operation contract defines the exact callback and receipt fields.

Reconciling unfinished work

Reconciliation asks a supported authoritative reader what happened to the original operation. It never invokes the effect callback again. Core waits until the original reservation's lease expires before reconciling pending work. If the process stopped after reserving the feature but before inserting its operation row, it can reconstruct that original pending row with the same definition, context, claim and token.

The reconciler may be another supported product that understands the frozen terms and can verify the authoritative original receipt. It reports confirmed applied, confirmed not applied, or unknown. Unknown retains the hold. An elapsed timeout does not prove absence: an earlier writer could have applied an effect before its response was lost.

Confirmed applied and confirmed not-applied results are terminal for that business reference in this bundled runtime. Repeating either result returns the retained receipt. There is no operation-reset or rearm control, attempt archive or automatic financial retry schedule. Keep the original feature, business reference, context and receipt for the owning integration's review. Creating another reference would create a different operation rather than resolve the first one.

If a terminal result could not release the feature's reservation, replaying that same original terminal result can safely retry the release. Do not clear a pending reservation or change revision numbers manually. A failed journal write can occur after the financial owner has already acted; use its authoritative evidence to resolve that uncertainty.

These are local integration services for compatible products. Free Gifts & BOGO does not offer shared-operation recovery buttons or execute its campaigns through a shared executor. The technical reconciliation contract describes the exact request, outcome and recovery boundaries.

Retiring a business policy safely

Business-policy retirement is a specific proof that an owner has no live policy left in an exact scope. It differs from retiring a shared executor. The two supported scopes concern current audience policy and protected-line policy. A compatible Suite host can expose B2B audience retirement controls. Free Gifts & BOGO shows accepted B2B removal proof and can use it only under its current promotion requirements; it has no retirement form. Host controls do not expose every owner/scope or release the B2B product's business freeze.

Use the owning product's preparation procedure before deactivating it. The product must establish a durable freeze and confirm no live policy remains. Core records the exact owner, scope, site, request identity, policy revision, evidence reference and freeze reference. Preparation also records the current runtime configuration fingerprint and a five-minute acceptance deadline. Core checks the proof's shape; the owner is responsible for the actual freeze.

Step	Requirement and effect
Prepare	Owner and its retirement guard are present. An authorized WooCommerce manager uses the current revision. The guard confirms no live policy and frozen state. Core writes Prepared evidence.
Deactivate owner	Follow the owner's supported procedure while keeping its freeze intact. Preparation alone does not remove the product.
Accept	Use a fresh request after deactivation, before five minutes expire. The guard and both site/network-active plugin entry must be absent, with the exact Prepared revision and unchanged runtime fingerprint.
Read accepted proof	A compatible consumer freshly verifies the accepted original record. It does not substitute the management display or an error.
Cancel	An authorized exact-revision write revokes the record. The owner separately decides when its own resume conditions are satisfied.

An attestation can freeze owner-side work before Core's final save succeeds. If preparation fails, keep the freeze and check the actual stored result; do not assume the owner can resume. Likewise, an acceptance notification can fail after the Accepted record was saved. Reread persisted state instead of repeating a different operation on the assumption of failure.

Accepted proof does not expire simply because the original five-minute preparation deadline passes. It remains subject to its site, runtime fingerprint, absent owner and unrevoked state. Changed runtime expectations, returning owner or invalid data make it unusable. Retirement does not erase old audience/protection requirements, transfer business prices, change historical order evidence or grant consumer discounts. Ordinary Core readers still hold when their required provider is absent unless a compatible consumer explicitly uses the separate retirement proof under its own rules.

The management view separates **stored state** from **validity**. State can be None, Prepared, Accepted or Revoked. Validity can be none, current, expired, stale, owner present, revoked or unavailable. Prepared can be current while the owner is present; that does not make acceptance possible yet. The owner-present flag reflects the collected guard or retained plugin basename, not an unrestricted scan of every possible owner.

Revocation reporting also distinguishes not attempted, confirmed and unconfirmed. This is what the current request observed; a later stored revision may differ. A read/write error must not be shown as confirmed revocation.

Cancellation always writes and advances the revision, even if the record was already revoked. Expired or stale proof can be cancelled, but corrupt or wrong-site records remain held for recovery. If no record exists, cancelling at revision zero creates a cancellation-only marker. That marker is not retirement proof; it prevents a delayed earlier preparation from appearing after cancellation. A later legitimate preparation requires the newer revision and new authentic owner evidence.

When an owner returns, Core attempts to revoke its previously accepted audience and protection records. It preserves Prepared records so the normal prepare → deactivate → accept sequence can work. Failed revocation remains unconfirmed: the owner's durable freeze must stay until all relevant cancellations succeed under its own write lock. A saved cancellation does not itself release that lock or resume business policy. Do not delete the journal or reset a revision to bypass a failed cancellation. The technical retirement contract contains exact fields and permission boundaries.

Discovering checkout credit participation

Core can advertise the currently registered Store Credit reservation participant to a compatible checkout coordinator. This discovery is independent of ordinary audience providers and shared feature owners. It does not reserve or spend credit.

The supported descriptor has exactly six values: schema 1, owner `affstackdev-store-credit`, capability `store_credit_reservation_participant_v1`, integer revision 1, and the owner's callable `enlist` and `order-verification` functions. Compatible plugins supply this descriptor in code during checkout-participant collection at initialization priority 40. There is no customer field for choosing a callback or entering an alternate protocol revision.

Registration must finish before another plugin resolves the participant. A duplicate attempt for the supported owner makes the participant conflicted for the request, even if the second submission is identical or follows a

malformed first attempt. A thrown collection callback discards partial registration. Records are not saved across requests, and stale unavailable callbacks are discarded rather than reused.

Connection status	Meaning and response
Registered	Descriptor and callable functions are available. The actual transaction still needs its owning products' checks.
Not ready	Collection has not finished; the integration must resolve later in initialization.
Missing	No accepted supported descriptor was registered; restore the owning integration.
Unavailable	Required function is not callable; restore the compatible implementation.
Conflict	Competing supported-owner registrations occurred; correct them before a fresh request.
Failed	Collection could not complete; investigate the registering service.
Site mismatch	The active Core belongs to another site; use the original supported site context.

Reading status does not invoke either financial function. Resolving a participant checks the exact owner, capability and integer revision, then returns a detached descriptor for the trusted caller. It does not validate an order's financial proof, verify payment, mark an order paid or provide a substitute participant. B2B owns coordination, Store Credit owns its ledger and receipt evidence, and both products' actual transaction contracts remain necessary. Free Gifts & BOGO does not register or call this participant. Its [customer-referral offer connection](#) supplies read-only promotion facts; that is separate from reserving or spending credit. The technical participant contract describes this discovery interface.

Shared operation troubleshooting

Symptom	Appropriate response
Record or runtime cannot be verified	Restore the original site's compatible runtime/storage. Preserve unfamiliar schema and corrupt evidence for recovery; do not overwrite it as empty.
Save conflicts or reports busy	Reload the current revision and review concurrent changes before saving.
No supported feature owner	Check the exact definition, product availability, enabled/archive state, adapter registration and current permissions. A displayed old owner is not a grant.
Evaluation or claim expired	Obtain a fresh current evaluation; do not edit the token or extend its expiry.
Operation pending or outcome unknown	Keep the original reference and context. Reconcile through its authoritative owner after the lease boundary; do not submit a new financial operation.
Confirmed not applied	Preserve the terminal original receipt. This bundled runtime replays the result and does not reopen that operation for another effect.
Retirement preparation fails	Keep the owner's freeze, inspect current proof/revision and complete the owning product's supported preparation.
Acceptance is unavailable	Confirm owner deactivation, five-minute preparation window, unchanged runtime and exact revision. Reread actual state after ambiguous results.

Symptom	Appropriate response
Cancellation or automatic revocation is unconfirmed	Keep the freeze and inspect persisted state under the owner lock. Cancel at the current revision; do not assume it succeeded.
Retirement was accepted but benefits remain held	Acceptance is separate evidence, not a customer eligibility grant or an automatic clearing of provider requirements. Follow the consuming product's supported policy.
Checkout participant is missing/conflicted	Restore the exact compatible owner and correct registration timing/duplicates. Discovery failure does not authorize bypassing credit checks.

Provide support with the original site, feature/business reference and current error code. Keep the original journal contents intact while the owning integration resolves the case. The technical error lookup maps every returned code to its specific recovery boundary.

Use the complete installed references

Free Gifts & BOGO installs two complete references for its installed version. This customer guide explains every supported setup and operation. The technical reference supplies exact fields, rules, interfaces, permissions, retained records, errors and recovery. Word and PDF editions of the customer reference are on affstackdev.com.

Download either manual

1. Open **Free Gifts & BOGO** → **Settings** in the native or Suite workspace.
2. In **Documentation**, choose **Download customer reference Markdown** or **Download technical reference Markdown**.
3. Save the attachment. You can open Markdown in a text editor, document reader or compatible model documentation system.

Your WordPress account needs WooCommerce management permission. The panel remains available when licensing pauses new promotion work. Downloading a manual reads a local installed file; it does not submit the settings form, enable an offer, change a cart or contact the license service.

If WordPress rejects an old link, reload the screen and use its current button. An unavailable-document message means the selector or installed file could not be accepted; use the provided link or reinstall the plugin from **My Account** → **Downloads** on affstackdev.com. A permission message requires the appropriate merchant account. A read failure requires readable installed files. If platform requirements prevent the normal workspace from opening, the customer reference is also on affstackdev.com.

Use the manuals with AffStackDev AI

The installed package identifies both manuals to a compatible AffStackDev AI inventory reader, including the audience and release each manual describes. AffStackDev AI 0.1.0.11a understands these explicit fields. The reader can retrieve the full references instead of relying only on a short product summary.

After installing an updated Free Gifts & BOGO release, refresh its entry in the inventory so the reader sees the current installed manuals. Inventory inclusion and document access remain controlled by the reader. Refreshing documentation does not install the product, create a promotion or grant the assistant permission to change the

store. Use the normal authorized product controls or the separately reviewed workflow in [Work with Free Gifts & BOGO through AI](#) for changes described in a manual.

Work with Free Gifts & BOGO through AI

With AffStackDev AI 0.1.0.12 or a later compatible release and Free Gifts & BOGO installed, you can describe promotion work in ordinary language and review the proposed changes before they are applied. Free Gifts & BOGO supplies its own executable controls; the assistant uses the same promotion rules, coupon validation and license permissions as the product workspace.

Your connected WordPress account needs the permissions for the task. Include Free Gifts & BOGO in the AI plugin workspace so its current actions and references are available. A license hold can prevent promotion changes while still allowing retained records, reports and permitted preferences to be inspected.

Review and apply a task

1. Describe the promotion, coupon or setting and the exact change you want. Use familiar product names, SKUs, category names or promotion titles.
2. Let the assistant inspect the current records and prepare the change. Preparation saves no product configuration.
3. Open the project's WordPress review link. Check the selected records, before/after values, publication state, customer-facing effects and notices.
4. Approve that exact project revision in WordPress, then explicitly apply it.
5. Review the saved result and the promotion or coupon link in the receipt.

An approval applies to the displayed revision. If a product, offer, license or other relevant setting changes before application, prepare and review a fresh proposal. The assistant cannot approve the project on your behalf.

Supported Free Gifts & BOGO tasks

Task	What you can ask the assistant to do
Find records	Search and filter promotions and coupons; identify products by name/SKU, categories, tags, product groupings, roles and policy pages.
Create or edit a promotion	Set the name, private staff description, public wording, excerpt, draft/published state, offer template, priority, combining rules and dates.
Configure qualification	Set spend or quantity thresholds, qualifying scope, minimum spend, product/category/tag/type/grouping selections and purchase-only exclusions.
Configure gifts	Set reward method/count, exact automatic product or variation, choice pool, value cap/basis, per-order limit and reward-only exclusions.
Build reward levels	Add, edit or remove levels and their thresholds, products, pools, quantities and value limits.
Set customer conditions	Configure customer access, roles, order history, campaign/customer limits, sale/virtual/downloadable behavior and coupon interaction.
Set storefront presentation	Choose product, catalog, cart and policy-page display switches; edit customer wording.

Task	What you can ask the assistant to do
Manage saved promotions	Duplicate as a disabled draft, enable/disable selected promotions, move to Trash, restore as a disabled draft or confirm permanent deletion. Restore a saved WordPress revision after reviewing its exact contents.
Change global preferences	Set exclusions, purchase/product defaults, combining and coupon-spend defaults, customer label, policy-page selection, display mode, refund warning and uninstall preference.
Create the policy page	Create and select the normal draft containing the Current Promotions block. Edit and publish that ordinary page through its WordPress owner.
Manage basic coupons	Create or edit code, description, type, amount, expiry, spend/usage limits, combining, shipping and status; enable/disable, Trash, restore or confirm deletion.
Preview and report	Run the same saved-promotion calculation with selected totals, quantity, coupon presence and local time. Read the native dashboard/report periods and paged results.
Manage license readiness	Inspect status, check now/recover pending activation, confirm deactivation, change the pre-activation environment, reset a moved/unreadable local identity or check protected updates. Enter credentials in the secure native form.

The existing chapters explain what each field means and when an offer is eligible. AI editing preserves fields you did not mention. To clear an optional description, date or selection, say so explicitly. Clearing the gift label restores **Free Gifts**. A blank coupon code requests a generated native code that you can review before application.

Choose exact records and manageable changes

If two products or promotions have the same name, choose the intended record from the lookup results before preparing the task. A SKU is useful for a product or variation. Categories, tags and roles can use their ordinary names. A configured product grouping can use its taxonomy and term name. The assistant binds the selected records to the proposal and checks them again before saving.

Long selections are read in pages and changed in small batches. Existing exclusions remain in place unless you ask to remove or replace them. Long descriptions can be edited a passage at a time. If a combined task is too large for one review, the assistant prepares smaller projects so you can see each exact change.

For example: **“Create a disabled draft promotion called Weekend Mug Gift. Customers who spend 50 on qualifying purchases receive one of our Mug products automatically. Let me choose the exact Mug from lookup results, then show me the proposed settings for review.”** This task begins with a product choice and a disabled draft; inspect and preview it before separately publishing/enabling the offer.

Read results and recover an uncertain task

An **applied** receipt means the saved Free Gifts & BOGO result matched the reviewed outcome. Check the linked record for the final values. An enabled offer still needs a qualifying shopper/cart, current entitlement, available products and the normal checkout conditions; saving its settings does not award a gift immediately.

If a task reports **unknown**, inspect and reconcile its existing project. A timeout may have happened after a record or some bulk items were saved. Reconciliation observes the saved state and operation evidence; it does

not repeat the request. If the exact outcome cannot be established, the uncertainty remains for review. An overlapping change can remain blocked while the earlier operation is uncertain.

Bulk promotion changes and creating/selecting a policy page can have partial results. License checks and deactivation can depend on a server response. Reopening the project and native record helps establish what exists before you authorize any further action. Do not create a replacement project merely to retry an uncertain effect.

Advanced WooCommerce coupon options, store-credit balances, Core/Suite settings, ordinary page publication and financial/order operations retain their own controls. Ask their owning integration to prepare that work. Free Gifts & BOGO's action support does not itself authorize order payments, credit transfers, commissions, refunds or fulfillment.

Glossary

Term	Meaning
Promotion	A saved campaign with its own definition, identity, state and conditions.
Qualification	Evaluation of which purchases and customer conditions earn an offer.
Gift	A separately marked product or variation line awarded at zero price.
Automatic gift	A configured gift selected by the server when its requirements succeed.
Customer choice	An earned gift selection that the shopper explicitly fills from eligible candidates.
Candidate	A product or variation currently considered for a gift selection.
Reward level	A configured threshold with the rewards assigned to that reached level.
Slot	One earned selection position within an offer.
Value cap	A maximum reference value allowed for a gift; it is not a spendable balance.
Exclusion	A rule that prevents the affected purchase, gift or both from participating.
Stacking	The rules governing which promotions may participate together.
Audience	The customer classification and current eligibility supplied under an owning contract.
Reconciliation	Rechecking gift lines against current rules, purchased items and eligibility.
Attribution	Saved evidence identifying the promotion and reward associated with an order gift.
Campaign claim	Retained allowance allocation associated with an order and, where applicable, an account.
Revision	A version of a retained record used to reject stale changes.
Core	The bundled shared commerce runtime with its own version and supported contracts.
Retirement	Explicit release of an owning provider policy under the specified contract.
Nonce	A WordPress request token checked alongside the required user capability.