Catalog Health Scanner Developer Reference

Developers · https://wpfactory.com/docs/catalog-health-developers/

Every extension point in Catalog Health Scanner for WooCommerce Pro uses the `wpfchs_` prefix: filters for the check library, applicability, thresholds and meta keys, actions around scans and fixes, three custom database tables, and a small set of post meta keys. This page lists them as they exist in version 1.0.0, verified against the shipped source.

Classes use the `WPFCHS_` prefix, the text domain is `catalog-health-scanner-for-woocommerce`, and everything is reachable from the global accessor `wpfchs()`.

Getting at the plugin

`wpfchs()` returns the singleton main class. Its `core` property holds every subsystem:

Accessor	Class	Responsibility
`wpfchs()->core->checks`	`WPFCHS_Checks`	The check registry: all 70 definitions, category list, disabled list, and which checks are runnable and scored.
`wpfchs()->core->applicability`	`WPFCHS_Applicability`	Group states, auto-detection, and the store probes behind it.
`wpfchs()->core->issues`	`WPFCHS_Issues`	All reads and writes of the issues table, including ignore, restore, resolve, reopen, and scan comparison.
`wpfchs()->core->scanner`	`WPFCHS_Scanner`	Starting, stepping, pausing, and finishing scans, plus scan history.
`wpfchs()->core->scores`	`WPFCHS_Scores`	Score computation, bands, and historic recalculation.

`wpfchs()->core->profiles`	`WPFCHS_Profiles`	Built-in and custom scan profiles.
`wpfchs()->core->fixes`	`WPFCHS_Fixes`	Preview, apply, undo, and the fix log.
`wpfchs()->core->schedule`	`WPFCHS_Schedule`	Cron scheduling, digest, critical alerts, and post-scan auto-remediation.
`wpfchs()->core->export`	`WPFCHS_Export`	CSV export of a filtered issue list.
`wpfchs()->core->recommendations`	`WPFCHS_Recommendations`	The check to plugin recommendation map.
`wpfchs()->core->report`	`WPFCHS_Report`	The PDF audit report.
`wpfchs()->core->admin`	`WPFCHS_Admin`	Admin screens and the shared page shell. Only set in the admin.

Useful methods on the main class and on core:

Function	**Returns**
`wpfchs()`	The `WPFCHS` instance. Identical to `WPFCHS::instance()`.
`wpfchs()->version`	The plugin version string. Also available as the `WPFCHS_VERSION` constant.
`wpfchs()->plugin_url()` / `wpfchs()->plugin_path()`	Base URL and filesystem path, without a trailing slash.
`wpfchs()->has_wpfactory_admin_menu()`	Whether the shared WPFactory admin menu library is available to attach screens to.
`wpfchs()->core->get_capability()`	The capability required by every plugin screen and AJAX endpoint. Defaults to `manage_woocommerce`.
`wpfchs()->core->get_threshold( $key )`	A threshold value, falling back to the built-in default.

`WPFCHS_Check::severity_weight( $severity )`	The score weight of a severity: 4, 2, 1, or 0.5.

Filters

Core behaviour

Filter	Filters	Arguments
`wpfchs_capability`	The capability required for every screen, AJAX endpoint, export, and report download.	`$capability`
`wpfchs_checks`	The full array of check definitions before they are turned into check objects. The place to add, alter, or remove checks.	`$defs`
`wpfchs_categories`	The category ID to label map, in display order. The registry loads one definition file per category key.	`$categories`
`wpfchs_profiles`	All scan profiles, built-in and custom, after custom ones are merged in.	`$profiles`
`wpfchs_threshold_defaults`	The default values of every threshold, before the stored option is applied.	`$defaults`
`wpfchs_threshold`	One resolved threshold value, after the option and defaults have been considered.	`$value, $key`
`wpfchs_reset_option_keys`	The option keys deleted by Reset & re-run setup.	`$option_keys`

Applicability

Filter	Filters	Arguments
`wpfchs_applicability_resolve`	The resolved state of a group: `applicable`, `scored`, `reason`, `source`. The final word on whether a group runs and counts.	`$result, $group`

`wpfchs_applicability_auto_detect`	The auto-detected verdict for a group: `applicable` and `reason`. Only consulted when the group is set to auto.	`$result, $group`
`wpfchs_dimensionless_shipping_methods`	Shipping method IDs assumed never to depend on weight or dimensions. Defaults to flat rate, free shipping, local pickup, and pickup location.	`$method_ids`
`wpfchs_feed_plugin_signals`	The class names and constants used to detect an active product feed plugin.	`$signals`

Meta keys and taxonomies the checks read

These let you point Catalog Health Scanner at the fields your store actually uses, without touching the checks themselves.

Filter	Default value	Used by
`wpfchs_feed_meta_keys`	`array( '_global_unique_id', '_alg_ean', '_wpm_gtin_code' )`	`gtin_missing`, and feed group auto-detection.
`wpfchs_cog_meta_keys`	`array( '_alg_wc_cog_cost', '_wc_cog_cost' )`	The three cost and margin checks, the cost fixer, and cog group auto-detection. The first key in the array is the one the fixer writes to.
`wpfchs_condition_meta_keys`	`array( '_woosea_condition', '_woo_feed_condition', '_wpfchs_condition' )`	`condition_missing`.
`wpfchs_google_category_meta_keys`	`array( '_woosea_google_product_category', '_woo_feed_google_category', '_wpfchs_google_category' )`	`google_category_missing`.

`wpfchs_faq_meta_keys`	`array( '_wpfchs_faq', '_yoast_wpseo_faq', 'product_faqs' )`	`faq_missing`, both for the check and for deciding whether the store uses FAQ content at all.
`wpfchs_brand_taxonomies`	`array( 'product_brand', 'pwb-brand', 'yith_product_brand' )`	`brand_missing` and the brand fixer. The first registered taxonomy in the list wins.
`wpfchs_placeholder_phrases`	The lorem-ipsum and WooCommerce sample phrases	`content_placeholder`.

Scanning, fixing, and recommendations

Filter	Filters	Default
`wpfchs_scan_batch_size`	How many products a scan loads per database page.	25
`wpfchs_ajax_step_budget`	Seconds one browser-driven scan step may spend working before returning progress. Cron-driven steps use a fixed 20 seconds.	8.0
`wpfchs_generated_sku`	The SKU the generator produces for a product or variation. Arguments: `$sku, $product`.	Uppercased slug or parent SKU, plus the object ID
`wpfchs_recommendations_map`	The check ID to recommendation map behind the single dismissible line under an issue group.	See below

The recommendation map ships deliberately small: a recommendation only appears where the recommended plugin genuinely solves the finding. As of 1.0.0 it covers `gtin_missing`, the three cost and margin checks, and the three tax checks. Each entry is an array of `plugin`, `url`, and `text`.

Actions

Action	Fires	Arguments
`wpfchs_core_loaded`	At the end of the core constructor, once every subsystem is available on `$core`. The right place to register your own filters.	`$core`
`wpfchs_admin_loaded`	After the admin classes load, in admin requests only.	`$core`
`wpfchs_scan_started`	Immediately after a scan row is created, before any product is examined.	`$scan_id`, `$profile_id`, `$trigger`
`wpfchs_scan_completed`	After the last batch, once stale issues are resolved, the score is stored, and product counters are synced.	`$scan_id`
`wpfchs_auto_remediated`	After unattended post-scan auto-fixing actually changed something.	`$scan_id`, `$result`
`wpfchs_fix_applied`	After a fix batch is written and logged.	`$log_id`, `$check_id, $items`
`wpfchs_fix_undone`	After a fix batch is reverted and its issues reopened.	`$log_id`, `$check_id, $items`
`wpfchs_before_version_update` / `wpfchs_version_updated`	Around the stored version being bumped, after the tables have been created or updated.	`$version`

`$trigger` is either `manual` or `scheduled`. `$items` is an object ID to `{ before, after }` map, which is exactly what undo replays.

Cron events

Hook	Arguments	Purpose
`wpfchs_scheduled_scan_event`	None	Recurring event created from your schedule settings. Starts a scan, then hands off to the stepper.
`wpfchs_scan_step_event`	`$scan_id`	Single event that advances a running scan for up to 20 seconds and reschedules itself 30 seconds later until the scan completes.

Adding your own check

Hook `wpfchs_checks` and push a definition array. A definition understands the keys below; anything you leave out takes its default.

Key	Type	Default	Meaning
`id`	string	Empty	Unique check ID. A definition with no ID is discarded.
`category`	string	Empty	One of the ten category IDs. Definitions loaded from the shipped files get this automatically; a definition you add through the filter must set it itself.
`label`	string	Empty	The human label shown in the UI, the digest, and the report.
`explanation`	string	Empty	One plain-language line about the commercial impact.
`severity`	string	`'medium'`	`critical`, `high`, `medium`, or `low`.
`group`	string or null	null	Applicability group. Null means "inherit from the category", which maps inventory, shipping, tax, downloads, and feed to groups of the same name and leaves every other category always applicable. An empty string forces always applicable.
`pass`	string	`'product'`	`product` for a per-product test, or `catalog` for a single cross-product pass.

`applies`	callable or null	null	Receives the product and returns whether the check should run on it at all. Products it rejects are not counted in the denominator.
`check`	callable or null	null	The test itself. Return false when the product passes, or the offending value when it fails. A catalog pass takes no arguments and returns an object ID to `{ product_id, value }` map.
`fix`	string or null	null	The fixer key. Fixers are implemented in `WPFCHS_Fixes`, so a custom key needs your own handling.
`fix_type`	string	`'manual'`	`auto`, `bulk`, or `manual`.

```
add_filter( 'wpfchs_checks', function ( $defs ) {
□$defs[] = array(
□□'id'          => 'my_missing_care_label',
□□'category'    => 'content',
□□'label'       => __( 'Missing care label', 'my-plugin' ),
□□'explanation' => __( 'Garment pages without a care label generate avoidable returns.', 'my-plugin'
),
□□'severity'    => 'medium',
□□'applies'     => function ( $product ) {
□□□return $product->is_type( array( 'simple', 'variable' ) );
□□},
□□'check'       => function ( $product ) {
□□□$value = get_post_meta( $product->get_id(), '_care_label', true );
□□□return ( '' === $value ? __( 'No care label', 'my-plugin' ) : false );
□□},
□);
□return $defs;
} );
```

Your check appears in the settings tick boxes, in custom profile builders, in the category tab, in exports and the PDF report, and in the score at its severity weight, exactly like a shipped one.

Database tables

Three tables are created on activation and on every version update, all with the site table prefix. None of them hold customer data.

Table	Holds	Notable columns
`{prefix}wpfchs_scans`	One row per scan, running or complete. The source of the dashboard, the history screen, and every recalculation.	`profile`, `status` (running, paused, cancelled, complete), `started_at`, `completed_at`, `products_total`, `products_scanned`, `last_product_id` for resuming, `issues_found`, `score`, `score_data` (JSON: per-check counters, category points, trigger, skipped count, recalculation markers), `check_ids` (JSON list of the checks that ran).
`{prefix}wpfchs_issues`	The lifecycle of every finding across scans. One row per object and check, enforced by a unique key.	`object_id` (product or variation), `product_id` (the parent), `check_id`, `category`, `severity`, `status` (open, resolved, ignored), `issue_value` snapshot, `first_seen_scan`, `last_seen_scan`, `last_reopened_scan` for regressions, `first_seen`, `resolved_at`, `ignored_by`, `ignored_at`.
`{prefix}wpfchs_fixlog`	One row per applied fix batch, which is what makes undo possible.	`user_id`, `check_id`, `fixer`, `created_at`, `items` (JSON object ID to before and after values), `items_count`, `undone`, `undone_at`.

The tables are not dropped on deactivation or uninstall, so history survives a plugin update or a temporary deactivation.

Post meta keys

Meta key	Written to	Value	Purpose

`_wpfchs_open_issues`	Products	Integer	Cached count of open issues, rebuilt after every scan and updated after every fix, ignore, restore, and undo. Powers the sortable Health column on the products list. Deleted rather than set to zero when a product is clean.
`_wpfchs_last_restock`	Products and variations	Unix timestamp	The last time stock was set to a positive quantity, recorded from the WooCommerce stock hooks. Collected as history for checks that need it; no shipped 1.0.0 check reads it yet.

Transients

Transient	Lifetime	Holds
`wpfchs_autodetect`	1 hour	The auto-detected verdict of every applicability group. Cleared on settings save and after every completed scan.
`wpfchs_schema_probe`	1 day	Whether one representative product page emits Product JSON-LD. Cleared with the auto-detect cache.
`wpfchs_urlchk_` plus an MD5 of the URL	12 hours	The HTTP status of a remote download URL, so `download_url_error` does not re-request the same file on every scan.

AJAX endpoints

All admin AJAX actions are nonce-protected and require the capability from `wpfchs_capability`. They are listed for debugging; they are not a public API and their payloads may change between releases.

`wpfchs_scan_start`, `wpfchs_scan_step`, `wpfchs_scan_control`, `wpfchs_group_products`, `wpfchs_ignore_issue`, `wpfchs_restore_issue`, `wpfchs_ignore_bulk`, `wpfchs_restore_bulk`, `wpfchs_fix_preview`, `wpfchs_fix_apply`, `wpfchs_fix_undo`, `wpfchs_fix_all_quick_wins`, `wpfchs_preview_quick_wins`, `wpfchs_dismiss_rec`.

Notes on writing extensions safely

- Register your filters on `wpfchs_core_loaded` or earlier. The check registry builds itself lazily on first use, and the built list is cached for the request.
- A per-product `check` callback runs once per product per scan. Keep it free of remote calls, or cache them in a transient the way the download URL check does.
- Catalog passes run once at the end of a scan and are scored against the total published product count, so return only genuine failures.
- Returning an empty string, `null`, or `false` from a check all count as a pass. Return a short offending value instead of `true` where you can: it is stored on the issue and shown in the UI, the CSV export, and the PDF report.

Related reading

- Check library: all 70 checks, for every shipped check ID
- Settings reference, for the options behind these filters
- How the health score is calculated

Need help?

If you are extending Catalog Health Scanner and a hook does not behave as expected, contact Catalog Health Scanner support with the hook name and a short code sample.



CATALOG HEALTH SCANNER FOR WOOCOMMERCE

See every feature and the pricing on the Catalog Health Scanner for WooCommerce plugin page.

View plugin →

Related articles

- What Catalog Health Scanner Does (and What It Does Not)
- Install Catalog Health Scanner and Run the Setup Wizard
- Catalog Health Scanner FAQ and Troubleshooting
- The Catalog Health Dashboard and Score
- Run Your First Catalog Scan