



**SOFTWARESILOS**

# **Product Fees Documentation**

PDF Manual

MODULE

productfee

LAST UPDATED

2026-03-29

# Table of Contents

*Product Fees Documentation · productfee*

- [Overview](#)
- [Overview](#)
- Getting Started
  - [Installation](#)
  - [Configuration](#)
  - [Assign Fees To Products](#)
  - [Create Fee Types](#)
- Features
  - [Display Locations](#)
  - [Fixed Vs Percent](#)
  - [Partial Invoices Refunds](#)
  - [Tax Calculation](#)
- Reference
  - [Database](#)
- Troubleshooting
  - [Common Issues](#)
  - [Upgrade From Legacy](#)

# Overview

*Product Fees Documentation · /productfee*

The Product Fees extension lets you add one or more fees to products and carry them through the full sales flow:

- [Quote / Cart / Checkout](#)
- [Order](#)
- [Invoice \(including partial invoices\)](#)
- [Credit Memo \(including partial refunds\)](#)
- [Emails and PDFs \(optional\)](#)

## Quick Links

### Getting Started

- [Installation](#)
- [Configuration](#)
- [Create Fee Types](#)
- [Assign Fees To Products](#)

### Features

- [Fixed vs. Percent Fees](#)
- [Tax Calculation](#)
- [Where Fees Are Displayed](#)
- [Partial Invoices & Refunds](#)

### Troubleshooting

- [Common Issues](#)
- [Upgrade From Legacy Versions](#)

### Notes

- This knowledge base is customer-facing and intentionally avoids database/SQL instructions.
- Use Product Fees admin workflows in Magento for setup and maintenance.
- If a required workflow is missing in your admin, contact support for the correct package/setup path.

# Installation

*Product Fees Documentation · </productfee/getting-started/installation>*

## Requirements

- Magento 2 installed and working
- Composer access to MageB2B packages

## Install

```
composer config bearer.repo.softwaresilo.io <token>
composer config repositories.softwaresilo composer
https://repo.softwaresilo.io/
composer require mageb2b/productfee:*
php bin/magento module:enable MageB2B_Productfee
php bin/magento setup:upgrade
php bin/magento setup:di:compile
php bin/magento cache:flush
```

If your setup requires static content deployment:

```
php bin/magento setup:static-content:deploy
```

## Uninstall

```
php bin/magento module:uninstall MageB2B_Productfee
php bin/magento cache:flush
```

# Configuration

Product Fees Documentation · </productfee/getting-started/configuration>

## Admin Configuration

Go to:

- Admin Panel -> Stores -> Configuration -> MageB2B -> Product fees



## General

### Enable product fees

- Config path: `productfee/general/enabled`

## Tax

### Apply tax to product fees

- Config path: `productfee/tax/apply_tax`
- If enabled, fees marked as taxable will be added to Magento tax calculation via "associated taxables".
- A taxable fee needs a tax class (see "Fee Types" / "Assignments").

### Include product fees in subtotal

- Config path: `productfee/tax/include_in_subtotal`
- If enabled, fees are added into the **Subtotal** totals segment (after Magento tax collector runs).
- If enabled, there is **no separate "Product Fees" total row** in cart/checkout totals (the fees are part of subtotal).

## Display

All display settings below use the same options:

- Do not display
- Display excluding tax
- Display including tax
- Display both excluding and including tax

### Display in cart and checkout

- Config path: `productfee/display/cart_checkout`
- Controls the per-item fee breakdown (cart line item, mini summary item, etc.).
- Note: the cart/checkout totals row comes from the quote total segment `product fee` and is only shown when fees exist and "include in subtotal" is disabled.

## Display on product view page

- Config path: `productfee/display/product_view`
- Controls the fee rows shown on the product detail page.

## Display in sales documents

- Config path: `productfee/display/sales`
- Controls order/invoice/credit memo totals (frontend account pages, guest pages, and PDFs).

## Display in emails

- Config path: `productfee/display/email`
- Controls fee rows in order / invoice / credit memo emails.

## Hide zero fee rows from the totals section in admin

- Config path: `productfee/display/admin_totals_hide_zero_fee_rows`
- If enabled, admin order totals will not show the "Product fees" row when the calculated amount is `0.00`.

## Totals Sort Order

Magento total sort order:

- Config path: `sales/totals_sort/productfee` (default: `35`)

## ACL / Permissions

To grant access to Product Fees configuration in the Magento admin, allow:

- `MageB2B_Productfee::productfee_config`

# Assign Fees To Products

*Product Fees Documentation · </productfee/getting-started/assign-fees-to-products>*

Assign fee types to products for the relevant website scope.

## Scope

Assignments are website-scoped (not store-view scoped). Always verify you are configuring the intended website.

## Recommended Admin Workflow

1. Open the product in Admin
2. Go to the Product Fees section for that product (or equivalent fee assignment area in your setup)
3. Add one or more fee types
4. Set the amount per fee type:
  - fixed amount for fixed fees
  - percent value for percent fees
5. Optionally set a tax class override where supported
6. Save product changes

## Editing / Removing Assignments

- Update the amount in the same Product Fees section
- Remove fee types that are no longer needed
- Re-test in storefront cart/checkout after changes

## Validation Checklist

- Product belongs to the intended website
- Fee type is active
- Amount is greater than zero where expected
- Display settings are enabled for target locations

## Next

- [Where Fees Are Displayed](#)
- [Common Issues](#)

# Create Fee Types

*Product Fees Documentation · /productfee/getting-started/create-fee-types*

Create reusable fee definitions in Magento Admin. Depending on your package/setup, this is done either in a Product Fees fee-type screen or via product fee attributes.

## What A Fee Type Contains

A fee type defines:

- A unique code
- A customer-facing label
- Calculation type: fixed or percent
- Taxable flag
- Optional default tax class
- Active status and sort order

## Fixed vs. Percent

- **Fixed:** amount per unit (for example 10.00 per item)
- **Percent:** percent of unit price (for example 5 = 5%)

## Taxable Fees

Tax is applied only when all of these are true:

1. **Apply tax to product fees** is enabled in configuration
2. The fee type is marked as taxable
3. A valid tax class is available for that fee context

## Recommended Admin Workflow

1. Open Magento Admin
2. Use your fee-definition area:
  - Product Fees fee-type screen (if available), or
  - **Stores -> Attributes -> Product** for fee attributes in legacy-style setups
3. Define code, label, amount type, taxable flag, and sort order
4. Save and keep the fee definition active
5. Test with one product before broad rollout

## Best Practices

- Use stable codes (handling, service, environmental)
- Keep labels customer-readable
- Start with a small test set, then expand

## Next

- [Assign Fees To Products](#)
- [Tax Calculation](#)





# Display Locations

*Product Fees Documentation · /productfee/features/display-locations*

Product Fees can be displayed in multiple places. Visibility is controlled via:

- `productfee/display/cart_checkout`
- `productfee/display/product_view`
- `productfee/display/sales`
- `productfee/display/email`

## Product Page

- Fee rows are rendered on the product detail page.
- Screenshot placeholder: `resources/kb/productfee/_screenshots/product-page.png`

## Cart

### Per Item

- Fee rows can be shown below each cart item.
- Screenshot placeholder: `resources/kb/productfee/_screenshots/cart-item.png`

### Totals

- If `productfee/tax/include_in_subtotal = No`, a "Product Fees" totals row is shown.
- If `productfee/tax/include_in_subtotal = Yes`, fees are part of subtotal and there is no separate row.
- Screenshot placeholder: `resources/kb/productfee/_screenshots/cart-totals.png`

## Checkout

- Same behavior as cart totals (segment `productfee` is only displayed when not included in subtotal).
- Screenshot placeholder: `resources/kb/productfee/_screenshots/checkout-summary.png`

## Customer Account / Guest Order Pages

- Order totals: shown in `order_totals` if `productfee/display/sales` is enabled.
- Invoice and credit memo totals: shown similarly.
- Screenshot placeholder: `resources/kb/productfee/_screenshots/order-totals.png`

## Emails

- Per-item fee rows are added under each line item (if enabled).
- Totals row is also added (if enabled).
- Screenshot placeholder: `resources/kb/productfee/_screenshots/email-order.png`

## PDFs

- A "Product fees" total is added to Magento sales PDFs when enabled.
- Screenshot placeholder: `resources/kb/productfee/_screenshots/pdf-invoice.png`

# Fixed Vs Percent

*Product Fees Documentation · </productfee/features/fixed-vs-percent>*

Each fee type defines how its value is calculated:

## Fixed Fees

- `amount_type = 0`
- The assigned `amount` is used as a fixed per-unit fee.
- Total fee for the cart item is `amount * qty`.

## Percent Fees

- `amount_type = 1`
- The assigned `amount` is interpreted as a percentage.

Calculation details:

- In cart/quote totals, the percent is calculated on the quote item's base unit calculation price.
- On the product detail page, the percent is calculated on the product's final price (display only).

Tip: if you use percent fees together with discounts, the effective base price used for calculation can differ based on Magento pricing rules and how the item price is calculated.

# Partial Invoices Refunds

*Product Fees Documentation · /productfee/features/partial-invoices-refunds*

Product Fees are stored on sales items and are automatically prorated by quantity for invoices and credit memos.

## Partial Invoices

When creating an invoice for a subset of ordered quantity, fee amounts are calculated as:

```
fee_on_invoice_item = fee_on_order_item * (invoiced_qty / ordered_qty)
```

The prorated values are written to invoice item fields:

- `base_productfee_amount`, `productfee_amount`
- `base_productfee_tax_amount`, `productfee_tax_amount`
- `productfee_applied_json`, `productfee_tax_json` (copied from order item snapshot)

## Partial Refunds (Credit Memos)

Credit memos use the same ratio logic:

```
fee_on_creditmemo_item = fee_on_order_item * (refunded_qty / ordered_qty)
```

This keeps fee totals consistent across multiple partial invoices / refunds.

# Tax Calculation

*Product Fees Documentation · </productfee/features/tax-calculation>*

## When Product Fee Tax Is Applied

Product fee tax is applied when:

- **Apply tax to product fees** is enabled, and
- the fee type is marked as taxable, and
- a valid tax class is available for the fee context

If any of these conditions is missing, the fee is treated as non-taxable.

## How Tax Is Calculated

Taxable fees are included in Magento tax calculation together with cart items. The resulting fee tax amounts are then carried through order, invoice, and credit memo documents.

## Include Fees In Subtotal

If **Include product fees in subtotal = Yes**:

- fee amounts are included in the **Subtotal** section, and
- there is no separate **Product Fees** total row in cart/checkout totals.

If **Include product fees in subtotal = No**:

- fees are shown as a separate **Product Fees** totals row in cart/checkout.

# Database

*Product Fees Documentation · /productfee/reference/database*

This page provides a high-level overview of how Product Fees data is structured.

## Fee Types

A fee type defines reusable metadata such as:

- internal code
- customer-facing label
- calculation type (fixed/percent)
- taxable flag
- active status and sort order

## Product Assignments

Fee types are assigned to products with website scope and amount values.

## Sales Snapshots

When a customer checks out, fee details are stored as snapshots on sales items so the same fee breakdown can be reused in:

- orders
- invoices
- credit memos
- emails and PDFs

## Why Snapshots Matter

Snapshots keep fee values stable for document history, even if fee setup changes later.

## Note

This customer-facing KB intentionally avoids SQL instructions. Use admin workflows for setup and maintenance.

# Common Issues

Product Fees Documentation · </productfee/troubleshooting/common-issues>

## Fees Do Not Show Anywhere

Checklist:

1. Module is enabled:
  - `php bin/magento module:status MageB2B_Productfee`
2. Extension is enabled in config:
  - `productfee/general/enabled = Yes`
3. Fee type is active in admin
4. Product fee assignment exists for the correct website
5. Assigned amount is greater than zero

## Fees Show In Cart Item, But Not In Totals

- If `productfee/tax/include_in_subtotal = Yes`, there is no separate "Product Fees" totals row.
- In that mode, fees are part of subtotal.

## Fees Show In Totals, But Not Under Cart Items

- Check `productfee/display/cart_checkout`:
  - If set to "Do not display", the per-item breakdown block will not render.

## Fee Tax Is Always 0.00

Checklist:

1. `productfee/tax/apply_tax = Yes`
2. Fee type is marked as taxable
3. A valid tax class is available for the fee context

## Emails: Totals Row Shows, But Per-Item Fee Rows Are Missing

- Per-item rows in emails are controlled by `productfee/display/email`.
- Totals row visibility in sales documents is controlled by `productfee/display/sales`.
- Verify both settings are configured as intended.

## Percent Fee Differs Between Product Page And Cart

In some setups, percentage presentation can differ by context (product page vs cart calculation base). Validate with a real product/cart test and ensure pricing configuration is consistent across websites.



## **Multi-Website Setup: Fees Missing On One Store**

- Fee assignments are website-scoped.
- Ensure assignments exist for each website that should show fees.

# Upgrade From Legacy

*Product Fees Documentation · </productfee/troubleshooting/upgrade-from-legacy>*

Older Product Fees versions may use a different setup model than current versions.

## If You Are Upgrading

During upgrade, migration logic may convert legacy fee data and settings into the current format used by the extension.

## Legacy Setup Indicators

If your store used an older setup, you may still see legacy fee-related product attributes or old fee labels in historical data.

## After Upgrade

- Runtime fee calculation uses the current Product Fees data model.
- Existing sales documents keep historical values for reporting consistency.

## Migration Did Not Recreate Expected Fee Setup

Common reasons:

- legacy source data is incomplete
- legacy prefixes/settings were not available
- products had no effective fee values

If this happens:

1. Recreate fee types in Admin
2. Reassign fees to target products/websites
3. Re-test cart, checkout, order, invoice, and credit memo flows

See:

- [Create Fee Types](#)
- [Assign Fees To Products](#)