Upgrade from Legacy Versions

Product Fees Documentation

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-09-26

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Product Fees

Upgrade from Legacy Versions

Current versions store reusable fee definitions in `productfee_type` and website-specific product assignments in `productfee_product_fee`. Older installations stored fee amounts, percentage flags, and tax values in product attributes.

What setup:upgrade migrates

The data patch reads the configured fee, percentage, and tax attribute prefixes. For each matching fee attribute, it:

1. derives a fee code from the attribute suffix;
2. creates a reusable fee type when it does not exist;
3. reads the product value in each website's default store, with global values as fallback;
4. creates or updates the website-specific product assignment;
5. carries the percentage flag and tax-class value into the new records when those source attributes exist.

Percentage fees use a separate type code ending in `_percent`. Zero and negative source amounts are not migrated.

The configuration patch also maps the earlier display settings to the current cart, product, sales, email, and tax settings. Note that it **forces three values**: it enables the extension (`enabled`), switches on tax application (`apply_tax`) and disables inclusion in the subtotal (`include_in_subtotal`). Review those settings after `setup:upgrade` if your shop needs a different tax treatment. Magento records both patches in `patch_list`, so they do not rerun on every deployment.

Check the source before upgrading

Record the three configured attribute prefixes before running `setup:upgrade`. If any prefix is empty, or no product fee attribute matches the fee prefix, the assignment migration has no source data to process.

For each website, keep examples of:

- a fixed fee;
- a percentage fee;
- a fee with a tax-class value;
- a product with a store-specific value;
- a product that should have no fee.

Verify the result

1. Open **Catalog > Product Fee Types** and confirm the expected type codes and labels.

2. Open each example product and check its website-specific **Product Fees** rows.
3. Test cart and checkout totals on every affected website.
4. Create an order, invoice, and credit memo, then compare their stored fee breakdowns.
5. Keep historical sales documents unchanged. They retain their item-level fee snapshots.

If an assignment is missing, first check the old prefixes, attribute values, website default store, and whether the source amount was greater than zero. Recreate a missing assignment through the Product Fees Admin fields rather than direct SQL.