



WooCommerce BACS Extended Version 1.0.1

User Documentation

TABLE OF CONTENTS

Overview	3
What Is BACS-X?	3
How the Plugin Works	4
Quick Setup	5
Install	5
Upload and Activate	6
Upload	6
Activate	6
Gateway Configuration	7
The Reconciliation Workflow	9
Importing CAMT.053	10
Importing CSV & Mapping Profiles	11
Create a Profile	11
Matching Rules	13
Booking (Marking as Paid)	14
Compatibility (HPOS, Blocks)	15
Data Storage	16
Hooks for Developers	17
Filters	17
File Map	17
Troubleshooting	18
Requirements & Compatibility	19
Changelog	20
1.0.1 – 2026-07-02	20
1.0.0 – 2026-06-30	20

OVERVIEW

Direct Bank Transfer (BACS) for WooCommerce is an extended bank-transfer (BACS) payment gateway with a built-in payment reconciliation workflow: import your incoming bank payments from CSV or CAMT.053 and match them automatically against the open orders that were paid with this gateway.

Customers pay by bank transfer using their order number as the reference; you then import the incoming payments of your bank account and the plugin matches them against the open orders, so you can mark them as paid — per order or all at once. It is fully HPOS compatible and works in the classic and the block checkout.

Version	1.0.1
Author	Ralf Fuhrmann
License	GPL-2.0-or-later
Requirements	WordPress 6.9+, WooCommerce 8.0+ (tested up to 10.9), PHP 7.4+
Statement formats	CAMT.053 (ISO 20022 XML) and CSV with a flexible column mapping
Languages	English and German (fully translated)

WHAT IS BACS-X?

BACS-X is a copy of the WooCommerce core “Direct bank transfer” gateway, so the checkout behaviour is familiar: the order is placed on-hold until the money arrives. The difference is everything that happens after the order: instead of checking every transfer by hand, you import your bank statement and the plugin reconciles the incoming payments against your open orders.

Optionally, BACS-X can also reconcile and book orders paid with the built-in WooCommerce “Direct bank transfer” gateway (id bacs), so existing BACS orders are covered without switching payment method.

HOW THE PLUGIN WORKS

- Checkout — the customer chooses bank transfer; the order is set to on-hold and the order number is stored as the payment reference.
- Transfer — the customer transfers the money to your bank account using that reference.
- Import — you export the incoming payments from your online banking as CAMT.053 or CSV and import the file under WooCommerce → BACS-X Reconciliation.
- Matching — the plugin matches the payments to the open orders by reference and amount.
- Booking — you review the list and mark the orders as paid, per order or all matched at once.

QUICK SETUP

INSTALL

- Copy the woocommerce_payments-bacs folder to wp-content/plugins/.
- Make sure WooCommerce is already installed and active.
- Activate "Direct Bank Transfer (BACS) for WooCommerce". On activation the plugin creates its payments table ({prefix}wc_bacsx_payments).
- Enable and configure the gateway (see below).

Configure the gateway under WooCommerce → Settings → Payments → Bank transfer (BACS-X) (see below).

UPLOAD AND ACTIVATE

UPLOAD

After you have downloaded the 'Direct Bank Transfer (BACS) for WooCommerce' plugin, you need to upload it to your WordPress installation.

Please go to your 'Plugins / Add New' menu, select 'Upload Plugin' from the top links and choose the downloaded ZIP file at the upload field.

After selecting the file, please press the "Install Now" button.

ACTIVATE

After the plugin is uploaded you will get a success screen.

Please use the link "Activate Plugin" to activate the plugin.

Setting	Description
Enable/Disable	Turn the gateway on/off.
Title / Description / Instructions	As in core BACS — shown at checkout, on the thank-you page and in emails.

Account details	One or more bank accounts (name, number, bank, sort code, IBAN, BIC) shown to the customer.
Status after booking	Order status set when a payment is booked: Payment complete (recommended), Processing or Completed.
Amount tolerance	Allowed difference between the order total and the incoming amount when matching (e.g. 0.00 for an exact match).
Standard WooCommerce BACS	Also match and book incoming payments against orders placed with the built-in WooCommerce Direct bank transfer (BACS) gateway (id bacs), not only this plugin's gateway. Off by default.

The customer is always shown their order number as the payment reference; matching relies on the customer using it.

THE RECONCILIATION WORKFLOW

Open WooCommerce → BACS-X Reconciliation. The page has three parts:

- **Import incoming payments** — upload a CAMT.053 or CSV file (and pick a CSV profile for CSV). On import the file is parsed, the payments are stored (duplicates are skipped) and automatically matched.
- **Open orders & assigned payments** — every open BACS-X order with the payment that was matched to it (amount, value date, reference) or a “not matched” hint. Each row has a Book as paid button, plus Book all matched payments and Re-run matching buttons.
- **Imported payments** — all imported payments with their status (unmatched / matched / booked), the assigned order, a field to assign a payment to an order manually, and a Remove action.

 Screenshot: The “BACS-X Reconciliation” admin page

BACS-X Reconciliation

Import incoming payments

Statement file

Datei auswählen

Keine ausgewählt

Format

Auto-detect ▼

CSV profile

— only needed for CSV — ▼

Import & match

IMPORTING CAMT.053

CAMT.053 (ISO 20022 BankToCustomerStatement) has a fixed structure. The parser:

- is namespace agnostic — it uses local-name() XPath, so it works with camt.053 .02 / .08 / .14 and also with files that declare the ISO namespace only as a prefix (a common real-world quirk),
- imports only incoming credits (CdtDbtInd = CRDT),
- reads the amount and currency, the booking/value date, the remittance information (RmtInf/Ustrd, plus AddtlntryInf and the reference ids as fallback), and the payer name/IBAN.

Just upload the .xml file; the format is auto-detected.

A sample file (11_EBICS_camt.053_...xml) and the schema (camt.053.001.14.xsd) ship with the plugin for reference.

IMPORTING CSV & MAPPING PROFILES

CSV exports differ from bank to bank, so you describe the structure once as a profile and reuse it.

CREATE A PROFILE

Under WooCommerce → BACS-X CSV Profiles:

- (optional) Analyse a sample CSV — upload a sample and the editor shows the columns with their position numbers (Pos 1, Pos 2, ...) and example values.
- Fill in the field mapping — the column position (1 = first column, 0 = not used) for:
 - Reference / purpose
 - Amount
 - Date
 - Currency (optional, defaults to EUR)
 - Payer name
 - Payer IBAN
- Set the delimiter, enclosure, whether the first row is a header, the decimal/thousands separators and the date format (PHP format, e.g. d.m.Y).
- Save the profile.

 Screenshot: The CSV mapping profile editor with sample-column analysis

BACS-X CSV Profiles

A CSV profile describes how a bank CSV export is structured: which column holds the amount, the reference, the date and so on. Profiles are reused when you import

Existing profiles

Name	Delimiter	Reference
No profiles yet.		

Add profile

Analyse a sample CSV

Upload a sample CSV to see the column positions and example values. Use the position numbers below to fill in the field mapping.

Keine ausgewählt
 Delimiter:

Field mapping

Profile name

Delimiter

Enclosure

When importing a CSV, choose the matching profile. Amounts are normalised to a float (thousands separator removed, decimal separator converted); only positive amounts (incoming) are imported.

Imported payments							
Value date	Amount	Reference	Payer	Source	Status	Order	Action
2024-03-13	100.000,00 €	Küßgarden Best. Scheck vom 28.02.2024 Scheck Nr. 115486 015800004601/000200002	Deutsche Bundesbank / 22772 Hamburg	CAMT	matched	#119	Order # Assign Remove
2024-03-13	80.000,00 €	Vbl2 pac009 RT02 nach DCS Überweisungsdurchfall mit Festkupon pac009-0uF5u0u05-00004 K024K031230000001010 015800001146/0119000004		CAMT	unmatched	—	Order # Assign Remove
2024-03-13	50.000,00 €	Vbl2 pac009 RT02 nach DCS Überweisungsdurchfall mit Festkupon K024K03135000000100 105000004224/0116200000	Testbank	CAMT	unmatched	—	Order # Assign Remove
2024-03-13	145.015,00 €	Überweisungsdurchfall mit Festkupon K024K03123000000101 0158000005/0119000003		CAMT	unmatched	—	Order # Assign Remove
2024-03-13	100.000,00 €	Einzahlungen 2000000011240513 105000001191/0016200000		CAMT	unmatched	—	Order # Assign Remove

BOOKING (MARKING AS PAID)

Booking an order calls `WC_Order::payment_complete()` with the payment reference, sets the configured status after booking, stores the BACS-X status as paid, adds an order note (amount, value date, source) and marks the payment row as booked.

- Book as paid (per order) — books a single order.
- Book all matched payments — books every order that currently has a matched payment.
- You can also book an order without a matched payment (manual override) using the per-order button.

COMPATIBILITY (HPOS, BLOCKS)

- HPOS / custom order tables — declared compatible; all order access uses the WooCommerce order CRUD and `wc_get_orders()`.
- Cart & Checkout Blocks — declared compatible; a minimal Blocks integration (`assets/js/blocks.js`) registers the gateway in the block checkout (BACS-X has no custom checkout fields).

DATA STORAGE

- Imported payments are stored in the custom table {prefix}wc_bacsx_payments (with a unique raw_hash to prevent duplicate imports).
- The order keeps _bacsx_payment_status (pending / paid) and _bacsx_payment_reference as order meta.
- CSV mapping profiles are stored in the option wc_bacsx_csv_profiles.

HOOKS FOR DEVELOPERS

FILTERS

- `woocommerce_bacsx_icon` — gateway icon URL.
- `woocommerce_bacsx_accounts` — the bank accounts shown to the customer.
- `woocommerce_bacsx_account_fields` — the account fields shown on the thank-you page / emails.
- `wcbacsx_reconciled_payment_methods` — the list of gateway ids that are reconciled (default `bacs_x`, plus `bacs` when the Standard WooCommerce BACS setting is on).

FILE MAP

File	Purpose
woocommerce_payments-bacs.php	Bootstrap: constants, activation, HPOS/blocks, gateway + column registration
includes/class-wcbacsx-gateway.php	The payment gateway (copy of BACS, extended)
includes/class-wcbacsx-importer.php	CSV + CAMT.053 parsing and CSV profiles
includes/class-wcbacsx-reconciliation.php	DB table, matching and booking
includes/admin.php	Admin pages and POST handlers
assets/js/blocks.js	Block checkout integration

TROUBLESHOOTING

- **No payments are matched.** Make sure customers use the order number as the reference, the imported amounts are positive, the currency matches, and the amount tolerance is set appropriately. Use Re-run matching after changing settings.
- **A CSV imports the wrong values.** Re-check the profile: column positions are 1-based, and the decimal/thousands separators and date format must match the file. Use Analyse a sample CSV to verify the positions.
- **A CAMT file imports nothing.** It must contain booked credit entries (Ntry with CdtDbtInd = CRDT). Outgoing debits are ignored on purpose.
- **The gateway does not show in the block checkout.** Ensure assets/js/blocks.js is reachable and you are on WooCommerce 8.0+.
- **A payment was imported twice.** It is not — duplicates (same source, date, amount, reference, payer) are skipped via the unique hash.

REQUIREMENTS & COMPATIBILITY

- WordPress 6.9+
- WooCommerce 8.0+ (tested up to 10.9)
- PHP 7.4+
- Bank statements exported as CAMT.053 (XML) or CSV
- Declares compatibility with WooCommerce High-Performance Order Storage (HPOS)

CHANGELOG

1.0.1 — 2026-07-02

- Added a Standard WooCommerce BACS option: the reconciliation (incoming payments) can now also match and book orders placed with the built-in WooCommerce BACS gateway (id bacs), not only this plugin's own gateway.
- New helper `wcbacsx_reconciled_payment_methods()` (filterable via `wcbacsx_reconciled_payment_methods`).

1.0.0 — 2026-06-30

- Initial release.
- Extended BACS payment gateway (separate gateway id `bacs_x`) with payment reference and status tracking.
- Payment reconciliation: import CAMT.053 (namespace-agnostic) and CSV (user-defined, reusable mapping profiles with a sample analyser).
- Automatic matching by reference + amount, manual assignment, per-order and bulk booking.
- HPOS and block-checkout compatible. German/English documentation.