



CUSTOMER HANDBOOK • BASEUP LABS

Everything the BaseUp Shopify Connector does, from the storefront to the warehouse

The Shopify Connector links one or more Shopify stores to Odoo 20 and keeps products, stock, orders, fulfillments and refunds aligned in both directions. Shopify keeps the storefront and the checkout. Odoo keeps everything behind it: the catalogue, the warehouses, the sale order, the delivery and the invoice, *without anyone re-keying an order*. This handbook is the complete feature list, in the order you will use it.

VERSION	PLATFORM	SHOPIFY API
20.0.1.0.0	Odoo 20	GraphQL 2026-04
DATA MODELS	LICENSE	PUBLISHER
7 stored	OPL-1	BaseUp Labs

CONTENTS

I	What you have bought
II	How the connector is built
III	Connecting a store
IV	Webhooks & real-time updates
V	Locations & warehouses
VI	Products: Shopify → Odoo
VII	Products: Odoo → Shopify
VIII	Variants, options & images
IX	Inventory & stock levels
X	Orders: Shopify → Odoo
XI	Draft orders & quotations
XII	Fulfillment & tracking
XIII	Refunds & cancellations
XIV	Customers & addresses
XV	Scheduled automation
XVI	Manual sync controls
XVII	Settings & parameters
XVIII	Logs & troubleshooting
XIX	Guided tours
XX	Access & security
XXI	New in 20.0 & upgrading from 19.0
XXII	Limits, requirements & licensing

Menu paths are shown in a bar like `Shopify ► Operations ► Sync Now`. A **NEW IN 20.0** mark flags behaviour that is new or changed in this edition; Chapter XXI lists every change in one place. Day-to-day procedures are in the separate Operator Guide; installation is in the Developer Setup Guide.

What you have bought

One module, everything included. It talks to Shopify's GraphQL Admin API (version 2026-04) for all business operations, and uses REST only for the two endpoints with no GraphQL equivalent: the connection test and webhook registration.

Store connection

OAuth with the Shopify Dev Dashboard; the access token and webhook secret fill themselves in.

Real-time webhooks

13 topics, each signature-checked, so an order lands in Odoo within seconds.

Product import

The full catalogue with variants, prices, SKUs and images; incremental day to day.

Inventory sync

Pull or push on-hand levels per warehouse, with a clear rule for which side wins.

Draft orders

Shopify draft orders arrive as quotations; Odoo orders go out as draft orders.

Refunds & cancellations

Refunds create draft credit notes; cancelled and voided orders cancel the sale order.

Scheduled sync

A 30-minute auto-sync and an hourly retry of products that failed.

Audit log

Every sync, webhook and failure recorded under Shopify ▶ Logs.

Several stores

Connect as many stores as you sell through; every mapping is scoped to its store.

Locations & warehouses

Shopify locations and Odoo warehouses paired both ways for multi-warehouse stock.

Product export

Create and update products in Shopify, optionally on every save.

Order import

Shopify orders become Odoo sale orders, discounts and shipping included, confirmed on payment.

Fulfillment

Validating a delivery fulfils the Shopify order, with tracking in both directions.

Customers

Partners and addresses created on import, without duplicating returning buyers.

Manual controls

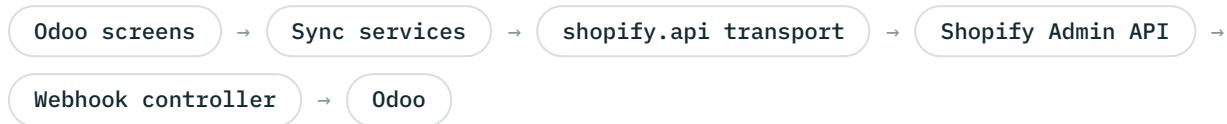
A direction-aware sync wizard plus per-store and per-record actions.

Guided tours

Four walkthroughs that drive the real screens for new users.

How the connector is built

Transport, business services and mapping records are kept strictly apart. No Odoo core app is modified; everything is added through inheritance inside this module, so Odoo upgrades and other apps are not disturbed.



MODEL	KIND	RESPONSIBILITY
shopify.account	Stored	Store credentials, configuration, per-store buttons, cron entry points
shopify.instance	Stored	Runtime handle passed into every sync call; webhook signature check; location resolution
shopify.product · .variant	Stored	Shopify product and variant IDs mapped to Odoo products, with sync state
shopify.order	Stored	Shopify order mapped to the sale order, with financial and fulfillment status
shopify.location.mapping	Stored	Shopify location ↔ Odoo warehouse pairs
shopify.log	Stored	The operation audit trail
shopify.api	Service	GraphQL requests, cursor pagination, staged file uploads, error mapping
shopify.product.sync	Service	Product import and export, variants, images, inventory push and pull
shopify.order.sync	Service	Order and draft-order import, fulfillment push, refund import
Sync & push wizards	Transient	The direction-aware manual sync, and product push with explicit store choice

Extended Odoo records

Products, product variants and attribute values, sale orders and lines, deliveries and stock quants gain Shopify fields and hooks, so Shopify status is visible on the records your team already works with.

Connecting a store

Each Shopify store is one account record. You supply the app's Client ID and Client Secret; Shopify's authorization flow supplies the rest.

Shopify ▶ Configuration ▶ Shopify Accounts

- 1 Enter the Shop URL (`mystore.myshopify.com`), Client ID, Client Secret and default Warehouse, then save.
- 2 Click **Connect to Shopify** and approve the requested scopes in Shopify.
- 3 Shopify returns to `/shopify/oauth/callback` , which exchanges the code for an access token. The token and webhook secret are stored and the status becomes **Connected**.

Callback checks

NEW IN 20.0

The callback verifies Shopify's `hmac` signature, refuses a callback naming a different shop, and sends the client secret only to the account's own store. The state token is single-use.

Requested scopes

Products, orders, inventory, fulfillments, merchant-managed fulfillment orders, draft orders (each read and write), plus read-only locations.

Public HTTPS address

OAuth and webhooks need Odoo reachable over public HTTPS. The system parameter `baseup_shopify.webhook_base_url` takes precedence over `web.base.url` , so a development server can keep a LAN address.

Test Connection

NEW IN 20.0

Calls Shopify's `/shop.json` . Success marks the account Connected and refreshes the shop name; failure shows a red notification and leaves the account in **Error**.

Secrets kept private

Access token, client secret, webhook secret, client ID and OAuth state are visible to Settings administrators only, so Sales Managers can run the connector without seeing them.

Per-store switches

Sync Products, Sync Orders and Sync Inventory on each account; new accounts take their defaults from Settings. Pricelist and Sales Team fields are available on the account.

Webhooks are what make the integration feel live: an order placed in Shopify appears in Odoo within seconds instead of waiting for the next scheduled run.

Shopify ▶ Configuration ▶ Shopify Accounts ▶ account ▶ Register Webhooks

TOPIC	EFFECT IN ODOO
orders/create	Import the order, confirming it when already paid
orders/updated	Refresh status and tracking; cancel the sale order when appropriate
orders/paid	Confirm the sale order
orders/cancelled	Cancel the linked sale order
orders/fulfilled , partially_fulfilled	Record fulfillment and validate the pending delivery
refunds/create	Create a draft credit note
products/create , /update , /delete	Update mapped products; flag a deleted product for clean-up
inventory_levels/update	Apply the new Shopify stock level at the mapped warehouse
draft_orders/create , /update	Create or update the draft quotation; promote it when completed
Clean registration	Registration deletes the app's existing webhooks on the store, then registers all 13 topics against your current base URL. It refuses <code>http://</code> , localhost and loopback addresses.
Signature check	Every request must carry a valid HMAC-SHA256 signature over the raw body, compared in constant time. A missing secret or bad signature is rejected with 401 before any handler runs.
Echo suppression	Pushing a product makes Shopify fire an update straight back. Unmapped products and updates no newer than our own push are skipped, so pushes do not bounce.
Concurrency	A webhook that collides with a scheduled sync is retried up to four times with backoff, each on a fresh snapshot. Handled failures return 200 and are written to the log, so Shopify does not retry in a storm.

Shopify counts stock per location; Odoo counts it per warehouse. A mapping table pairs them, which is what makes multi-warehouse inventory sync possible.

Shopify Accounts ▶ account ▶ Sync Locations · Push Warehouses to Shopify

Sync Locations

For each active Shopify location: skip it if already mapped, link it to an Odoo warehouse with the same name, or create a warehouse with a unique short code. Mapped locations are enabled for online orders.

Push Warehouses

Creates a Shopify location for every Odoo warehouse that has no mapping yet, using the warehouse address. Mapped warehouses are left alone.

Editable mapping table

Rows can be reordered and edited on the account form; each row pairs one warehouse with one Shopify location.

One pairing

each NEW IN 20.0

The database refuses a warehouse mapped twice on one account, or a Shopify location mapped twice, so the pairing is never ambiguous.

Single-location fallback

With no mapping rows, the account's own Warehouse is paired with the store's first active location. Explicit mappings are strongly preferred.

Product import picks its mode automatically per store: a full walk the first time, then only what changed.

Incremental import	Day to day, only products updated since the last sync are fetched, starting a configurable lookback window early so nothing falls through the boundary.
Full resync	The first sync, or Full Product Resync, walks the catalogue in product-ID order and saves its position, so a large catalogue resumes where each run stopped instead of starting again.
What an import creates	The product template, its attributes, values and variants matching Shopify's options, variant and inventory-item mappings, images unless skipped, and optionally stock levels.
Re-identification	Products first pushed from Odoo carry a tag naming their Odoo template, so a later import links them back instead of creating a duplicate.
Performance controls	A shared batch size, an option to skip image downloads during bulk runs, and bulk flags that stop an import from triggering pushes back to Shopify.
Errors do not stop a run NEW IN 20.0	A product that fails is marked Error and the walk moves past it; the hourly retry and Clean Up Errored Products handle it from there.

Pushing a product runs a fixed sequence, guarded by a per-product lock so two workers never export the same product at once.

1 · product → 2 · prices → 3 · stock → 4 · mappings → 5 · image

Field mapping	Name, sales description, vendor (company), product type (category) and, on creation only, a tag that identifies the Odoo template on later imports.
Self-healing	If a mapped Shopify product was deleted, the stale mapping is dropped and the product created again instead of failing forever.
Automatic pushes	Optional: push new products, and re-push linked products when name, description, category, image, active, internal reference, barcode or price changes. Both off by default; new-product push is skipped when several stores are connected.
Push to Shopify NEW IN 20.0	On the product form and as a list action (Action ▶ Push to Shopify). You choose the store; each product pushes in its own savepoint, and the notification reports real failures, turning orange when any product was rejected.
Attribute changes	Editing an attribute value's name, extra price, image or order, or a product's attribute lines, re-exports the affected products when auto-update is on.
Deleting a product	Archives its Shopify counterpart instead of deleting it, so Shopify's order history stays intact.
Two-way sync from the list	Sync Shopify Products pushes each selected linked product and then pulls it back, reporting pushed, pulled, skipped and failed counts.

VIII Variants, options & images

Matching a Shopify variant to the right Odoo variant is the hardest part of any Shopify integration. Both sides are reduced to the same key, such as `color=blue|size=xl`.

Matching order	Stored variant mapping, then option match within the linked template, then an on-demand product sync, then inventory item, then SKU, then title. A successful fallback is saved, so the slow path runs once.
Stale mappings repaired	A mapping whose options no longer match the order line is corrected in place, and a warning names both the old and the new variant.
Variant lifecycle	On export, variants missing in Shopify are created, variants removed in Odoo are deleted, and every price, including attribute extras, is pushed.
Images	Shopify's staged-upload flow: upload, attach, wait until the image is ready (up to two minutes), then delete the old image. Variant images are pushed per variant.

Inventory sync overwrites by design: whichever side is the source wins. Pull makes Shopify authoritative; Push makes Odoo authoritative. The scheduled sync defaults to Pull.

Push: Odoo → Shopify

For every warehouse and location pair, Odoo's available quantity is sent to Shopify. Unchanged quantities are dropped first, so a routine run sends almost nothing.

Safe writes

Each write carries an idempotency key and the quantity it expects to replace, so Shopify rejects a write based on a stale read.

Pull: Shopify → Odoo

Shopify's availability for every mapped variant overwrites Odoo's on-hand quantity at the mapped warehouse.

What triggers a push

Validating any transfer, editing an internal quant, a product export, the scheduled sync when inventory is enabled, or the sync wizard.

Reservations ignored

Reserving stock for an order is not a real movement and never pushes; a delivery pushes once, after the move is done.

Cron guards NEW IN 20.0

Inside a scheduled run, inventory push stops after 90 seconds or 300 items (both adjustable) and resumes next run. Quantities already collected are always sent before it stops.

Orders arrive by webhook within seconds, and the scheduled sync picks up anything missed, starting a lookback window before the last sync.

Shopify ▶ Operations ▶ Orders • Sales ▶ Orders

What an order becomes	A sale order for the resolved customer, with billing and shipping addresses, the Shopify order name as customer reference, the order note and date, and the store's warehouse and sales team.
Line provenance	Every line keeps the Shopify line ID, variant ID, a snapshot of the SKU at import time and the originating store.
Discounts & shipping NEW IN 20.0	Each line keeps its original price with a discount % equal to every discount Shopify allocated to it, order-level codes included. Each shipping line becomes a delivery line at the price the customer paid.
Auto-confirmation	An order already paid is confirmed straight away. If confirmation fails on routes or rules, the order stays in quotation and the reason is logged.
Status on the order	Shopify financial and fulfillment status show on the sale order, and the Shopify order name and status on its delivery.
No double imports	A per-order lock, a row check and a database unique constraint stop a webhook and a scheduled sync from importing the same order twice.

What is not imported

Shopify's own tax lines are not copied: Odoo applies taxes from your product and fiscal position settings. Shipping lines are added at Shopify's final price without Odoo tax on top.

Draft orders & quotations

Draft orders travel in both directions, so a quote prepared on either side reaches the other.

Shopify draft orders

Open draft orders arrive as Odoo quotations, labelled as drafts in the order list.

Completion NEW IN 20.0

When a draft order is completed in Shopify, the existing quotation is re-pointed at the real order and renamed from the draft name (D1234) to the order number (#1149). No second order appears.

Odoo orders to Shopify

A confirmed sale order for a connected store is exported as a Shopify draft order, built from the variant mappings and the customer's details. Export never blocks confirmation.

Respects your setting NEW IN 20.0

With **Sync orders by default** off, confirmed orders and quotations are not sent to Shopify.

Bulk export NEW IN 20.0

Action ▶ Sync Orders to Shopify and the two-way list sync report each order Shopify rejected, by name, instead of counting it as synced.

Fulfillment & tracking

The delivery is the source of truth for shipping: validating it in Odoo fulfils the order in Shopify, and a fulfillment in Shopify validates the delivery in Odoo.

Odoo → Shopify

Validating an outgoing delivery creates the Shopify fulfillment for the order's open fulfillment orders, with carrier and tracking number. Orders already fulfilled are skipped.

Shopify → Odoo

An `orders/fulfilled` webhook validates the pending delivery and writes the tracking number, matching the carrier by name when the delivery has none.

Hand-picked quantities kept NEW IN 20.0

If someone already picked part of the delivery in Odoo, that picked quantity is kept; untouched deliveries are validated in full.

Tracking updates

Changing the tracking reference on a done delivery updates Shopify, and tracking added later in Shopify updates the delivery. Each direction suppresses its own echo.

XIII Refunds & cancellations

Money flows back without anyone re-keying it: refunds become draft credit notes for finance to review, and cancelled orders close in Odoo.

Line refunds	A refund naming items creates a draft credit note against the posted invoice, with those lines, quantities, discounts and taxes, linked to the sale order lines so the order shows the credit note and its invoiced quantity drops NEW IN 20.0 .
Amount refunds	A shipping or adjustment refund without items becomes one credit-note line for the refunded transaction amount.
Safe repeats	Each credit note is tagged with its Shopify refund, so a re-sent webhook never creates a second one. Without a posted invoice the refund is logged for manual handling.
Cancellations	A cancelled order, or a voided and unfulfilled one, cancels the sale order. Orders Odoo refuses to cancel are logged, and a missed cancellation is repaired on the next sync.

Imported orders create real Odoo partners, without duplicating returning buyers.

Resolution order	Shopify customer ID stored in the partner reference, then email (individuals only, back-filling the ID), then a new partner from the Shopify name, email and phone.
Guest checkouts	Orders with no customer and no email attach to one shared Shopify Customer partner per company.
Addresses	Billing and shipping become child invoice and delivery addresses, reused when street, ZIP and city match. Country and state are matched by code or name.
Batch cache	A customer placing several orders in one run is looked up once.

Two scheduled actions keep stores in step when nobody is watching. Both are standard Odoo scheduled actions you can adjust or switch off.

Settings ▶ Technical ▶ Scheduled Actions

SCHEDULED ACTION	EVERY	WHAT IT DOES
Shopify: Auto Sync Connected Accounts	30 min	Products, then orders and draft orders, then inventory when enabled, for every connected store
Shopify: Retry Errored Products	1 hour	Re-attempts products left in Error, each in a fresh transaction
One store cannot undo another NEW IN 20.0		Each store syncs inside its own savepoint and is retried up to three times on a conflict. A store that fails is rolled back alone and logged; stores synced earlier in the same run keep their work.
Fits the time limit		Order and product runs stop after one batch and resume from their saved position next run, and calls use shorter timeouts inside scheduled runs.
Inventory is opt-in		Inventory is the slowest phase, so it only runs in the schedule once you enable it and choose its direction.

Anything the schedule does, you can trigger now, from the store, a list or a single record.

Shopify ▶ Operations ▶ Sync Now

Sync wizard	Choose the store, the direction (pull or push) and what to sync: all, products, orders or inventory. Batch Mode processes one batch and stops. Orders cannot be pushed; the wizard says so.
Store buttons	Connect, Test Connection, Register Webhooks, Sync Locations, Push Warehouses, Sync All, Full Product Resync and Clean Up Errored Products, plus Products and Orders counters.
Product actions	Push to Shopify and Pull from Shopify on the product form; Sync Shopify Products above the product list; Push to Shopify in the list's Action menu.
Order actions	Sync Orders to Shopify in the sale order list's Action menu, and Sync Shopify Orders above the list.

Settings ▸ Shopify Connector

SETTING	DEFAULT	EFFECT
Sync products / orders / inventory by default	On	Defaults for new accounts; orders also gates outbound order export
Run inventory sync in scheduled cron	Off	When off, the schedule handles products and orders only
Cron inventory direction	Pull	Pull = Shopify wins; Push = Odoo wins
Auto-push newly created products	Off	Export new products automatically (single store only)
Auto-update products on save	Off	Re-export linked products when watched fields change
Sync batch size	100	Shared batch size for products, orders and inventory
Skip product images in bulk sync	Off	Large speed gain on big catalogues
Order sync lookback window	1 hour	Overlap before the last sync date

SYSTEM PARAMETER	DEFAULT	EFFECT
<code>webhook_base_url</code>	—	Public HTTPS address for webhooks and OAuth; overrides <code>web.base.url</code>
<code>cron_inventory_max_seconds</code>	90	Time budget for inventory push in a scheduled run
<code>cron_inventory_max_items</code>	300	Item budget for inventory push in a scheduled run

Stored as real values NEW IN 20.0

All parameters live under the `baseup_shopify.` prefix. Odoo 20 stores an unticked box as a real “False”, so a setting you switch off stays off.

XVIII Logs & troubleshooting

Shopify ► Operations ► Logs

Every sync, webhook and failure is recorded with its store, type (success, info, warning, error), operation, record and Shopify ID, newest first.

SYMPTOM	CAUSE AND FIX
"Webhook URL must be a public HTTPS URL"	Set <code>baseup_shopify.webhook_base_url</code> to your public HTTPS address and retry.
"Shopify authentication failed"	The access token is invalid or revoked. Connect to Shopify again.
"Shopify rate limit reached"	Lower the batch size or lengthen the schedule interval.
Orders import but stay quotations	Confirmation failed on routes or rules; the reason is in the log.
Products stuck in Error	Let the hourly retry run, then use Clean Up Errored Products for products deleted in Shopify.
Inventory never finishes in the schedule	The 90-second / 300-item guard is working; raise it or run inventory from the wizard.
A new product did not auto-push	Auto-push is off, or several stores are connected. Use Push to Shopify.
Webhooks arrive but are rejected	The webhook secret does not match, or a proxy rewrites the body the signature covers.

Four interactive walkthroughs drive the real screens. Start one from the developer menu (Start Tour), or let the onboarding tour run for a new user.

TOUR	COVERS
<code>shopify_onboarding</code>	Creating the account, credentials, warehouse, connecting, registering webhooks, syncing locations
<code>shopify_first_product_push</code>	Pushing a first product to Shopify
<code>shopify_multi_warehouse_setup</code>	Mapping Shopify locations to Odoo warehouses
<code>shopify_first_order_review</code>	Reviewing an imported order and its linked sale order

Updated for Odoo 20 NEW IN 20.0

The tours follow the Odoo 20 web client, including its product kanban and “More” button menu, and run end to end in the module's automated tests.

RECORDS

SALES MANAGER

INTERNAL USER

Accounts, mappings, locations, logs

Read, write, create, delete

Read only

Sync and push wizards

Full

Full

Access rules NEW IN 20.0

Declared in Odoo 20's unified access rules, with exactly the same permissions as before. The Shopify menu is for Sales Managers; internal users keep read access so Shopify status stays visible on the products, orders and deliveries they work with.

Secrets

Access token, client secret, webhook secret, client ID and OAuth state are limited to Settings administrators and shown as password fields.

Webhooks

The webhook route is public by necessity but processes nothing before the HMAC check passes; an account with no webhook secret rejects every webhook.

OAuth

A 32-byte one-time state token, matched on callback and cleared immediately, plus Shopify's signature and a shop check NEW IN 20.0.

New in 20.0 & upgrading from 19.0

Version 20.0.1.0.0 is the Odoo 20 edition. Everything below is new or changed compared with 19.0.2.0.0.

AREA	WHAT CHANGED
Platform	Odoo 20 (Python 3.12+, PostgreSQL 16+); <code>stock_delivery</code> is now a declared dependency
Security	The OAuth callback verifies Shopify's signature and the shop; valid scopes only, including fulfillment orders
Orders	Discounts and shipping are imported, so the Odoo total matches Shopify
Draft orders	A completed draft order takes the real order number
Refunds	Credit notes are linked to the sale order lines
Data integrity	Unique pairings for products, variants, orders and locations are enforced by the database
Feedback	Failed pushes and exports are reported as failures; Test Connection keeps the Error state
Automation	One failing store no longer undoes others; the item limit sends what it collected; a failing product no longer stalls a full resync
Settings	“Sync ... by default” applies to new accounts; quotations respect “Sync orders by default”
Interface	Odoo 20 icons, access rules and updated guided tours; the product list action opens the push wizard
1	Back up the database and filestore, and test on a copy first.
2	Upgrade the database to Odoo 20 with Odoo's upgrade service.
3	Replace the module folder with the 20.0 version and run <code>-u baseup_shopify</code> . A pre-migration script removes duplicate mappings, keeping the newest, before the unique constraints are added.
4	In the Shopify app, replace <code>write_webhooks</code> with the two fulfillment-order scopes, then click Connect to Shopify again to grant them.
5	Click Register Webhooks if the public address changed, then place a test order and validate its delivery.

Limits, requirements & licensing

Taxes	Shopify tax lines are not imported; Odoo computes taxes from your configuration.
Orders are inbound	Odoo orders export as Shopify draft orders, never as completed orders.
Inventory overwrites	There is no merge; the configured direction wins outright.
Currency	Prices are read in the shop's own currency and booked in the Odoo company's currency, without conversion; use the same currency on both sides.
Refunds	Credit notes need a posted invoice; otherwise the refund is logged for manual handling.

REQUIREMENT	DETAIL
Odoo	20.0 (Python 3.12+, PostgreSQL 16+) with <code>sale_management</code> , <code>website_sale</code> , <code>stock</code> , <code>product</code> , <code>account</code> , <code>delivery</code> and <code>stock_delivery</code>
Shopify	A store and a Dev Dashboard app with the scopes in Chapter III
Network	A public HTTPS address for OAuth and webhooks; outbound HTTPS to Shopify
Scheduled actions	Odoo cron workers enabled for automatic sync

License

Module `baseup_shopify`, version 20.0.1.0.0, USD 630.00. Licensed under the Odoo Proprietary License v1.0 (OPL-1), published by BaseUp Labs. Support and implementation: support@baseuplabs.com.

BaseUp Shopify Connector · Bidirectional Shopify integration for Odoo

Module **baseup_shopify** · version **20.0.1.0.0** · Odoo 20 · Shopify Admin API 2026-04 · license **OPL-1**

© BaseUp Labs. Feature handbook compiled 2 October 2026 from the shipped module.