magmodules

Product Updates Configuration

Channable Connect for Magento 2

This guide explains how to set up real-time product updates for Channable Connect. When stock levels or prices change in Magento, these updates can be pushed to Channable automatically—keeping your marketplace listings accurate without waiting for the next full feed sync. This is especially useful for high-volume stores where inventory changes frequently.

Location: Channable → Product Updates

General

Enable

Turns the product updates feature on or off for this store view. When enabled, Channable can receive real-time stock and price updates.

Products per Run

Sets a limit on how many products are updated in each sync cycle. The maximum is 200.

Recommended: 50. This balances update speed with server performance.

Webhook

The webhook URL that Channable uses to fetch product updates. This field appears at the store view level when you've enabled product updates. Copy this URL into your Channable account to establish the connection.

Enabled Stores

Shows an overview of which store views have product updates enabled and their webhook URLs. This helps you verify your multi-store configuration at a glance.

Settings

Invalidation Type

Modus

Choose how product changes are detected:

- **Observer** - Detects changes in real-time as products are saved. Stock changes, price updates, and attribute modifications trigger immediate invalidation.

- **Cron** - Periodically scans for changed products. Less immediate but can be more reliable in some server environments.

When to use Observer:

- You need near-instant updates to marketplaces
- Your server handles event observers without performance issues

When to use Cron:

- Observer-based updates aren't triggering reliably
- You prefer predictable, scheduled updates over real-time

Cron

Enable

Turns on the cron-based sync process. When enabled, Magento will periodically send queued product updates to Channable.

Important: If disabled, no product updates will be sent to Channable, even if products are invalidated.

Frequency

How often the cron job runs to process queued updates:

- **Every 5 minutes**
- **Every 15 minutes**
- **Every 30 minutes**
- **Every hour**
- **Custom**

Recommended: Every 5-15 minutes for most stores. More frequent updates keep listings accurate but increase server load.

Custom Frequency

When "Custom" is selected, enter a cron expression for a custom schedule. Format: `* * * * *` (minute, hour, day of month, month, day of week).

Example: `*/10 * * * *` runs every 10 minutes.

Debugging

Log Calls

When enabled, all API calls and product updates are written to the Channable log file at `var/log/channable.log`.

When to enable:

- Troubleshooting why updates aren't reaching Channable
- Verifying which products are being updated
- Debugging webhook connection issues

When to disable:

- Production environments where logging overhead is a concern
- Once you've confirmed updates are working correctly

How Product Updates Work

1. **Product changes are detected** - Either through observers (immediate) or cron scanning (periodic)
2. **Products are queued for update** - Changed products are marked as "invalidated" in the queue
3. **Cron processes the queue** - The cron job picks up invalidated products (up to the "Products per run" limit)
4. **Updates are sent to Channable** - The webhook pushes stock and price data to Channable
5. **Channable updates marketplaces** - Channable processes the update and syncs with connected marketplaces

Need More Help?

Documentation:

- All Help Articles - Complete documentation overview

Support:

- Contact Support - Get help from our team

For a complete overview of features and configuration options, see the Channable Connect extension on magmodules.eu

All articles for Channable Connect

Installation

1	Installation using Composer (recommended)
2	Installation using the Adobe Marketplace
3	Install through FTP and SSH

Configuration

1	Quick Start Guide
2	Product Feed Configuration
3	Orders Configuration
4	**Product Updates Configuration (current)**

Troubleshooting

1	The price of at least one product is missing a currency
2	Test Feed
3	Test Orders

Grids

1	Returns Grid
2	Product Updates Grid
3	Orders Grid

Background

1	How does the Return process work

2	Best Practices
3	CLI Commands