
Table of Contents

Introduction	1.1
Requirements and dependencies	1.2
Installation	1.3
Configuration	1.4
Compatibility	1.5
Update	1.6
Store admin guide	1.7
Products	1.7.1
Order	1.7.2
Stock	1.7.3
Activity Log	1.7.4

Introduction

This manual will support you with setting up [YellowCube](#) plugin on your WooCommerce store. Only YellowCube specific topics are covered in this document. In order to efficiently set up the website you should be familiar with Wordpress and WooCommerce. If that is not the case we strongly suggest you study the [Wordpress](#) and [WooCommerce](#) documentation pages first.

This document is publicly developed on [GitHub](#) in the [GitBook](#) format. If you notice any mistakes or inconsistencies please consider proposing improvement by opening a [pull request](#) or [opening an issue](#) describing the problem.

If you need any support by the YellowCube team please contact us at www.yellowcube.ch or by phone [+41 58 386 48 08](tel:+41583864808)

Table of contents

- [Introduction](#)
- [Requirements and dependencies](#)
- [Installation](#)
- [Configuration](#)
- [Compatibility](#)
- [Update](#)
- [Store admin guide](#)
 - [Products](#)
 - [Order](#)
 - [Stock](#)
 - [Activity Log](#)

License

// TODO define license

Requirements and dependencies

1. *YellowCube account*: In order to use the YellowCube shipping services you will first need to obtain an account. In order to do that you'll need to get in touch with the Swiss Post by using the [online contact form](#). Post's staff will lead you through the onboarding process and provide you with the account information. This will include, among other information, things like *Sender ID*, *Receiver ID*, *Depositor Nr.*, *Partner Nr.* and *Plant ID*. You will need those when configuring the shop.
2. *Wordpress site with WooCommerce*: You will obviously need a Wordpress site with its eCommerce platform WooCommerce installed on it. If you are not sure how to do that consider checking the relevant documentation sections([\[1\]](#), [\[2\]](#)) or finding professional support. Wordpress, WooCommerce and WooYellowCube plugin are all open source software which means that there are many experts with the necessary expertise available.
3. *Server requirements*:
See also [WooCommerce requirements](#)).
 - *PHP version*: WooYellowCube plugin will work with any supported version of PHP 7+.
 - *SOAP PHP extension*: YellowCube uses [SOAP](#) to communicate with individual stores. WooYellowCube plugin requires [PHP SOAP extension](#) to be installed on the server for that reason. If you are unsure if that is the case you should check with your hosting provider.
 - *MySQL database*: MySQL version 5.6 or greater OR MariaDB version 10.0 or greater is required.
4. *HTTPS support*: eCommerce websites usually deal with potentially sensitive information. In order to protect those we **strongly** suggest using HTTPS encryption. If you are not sure where to get the certificates we suggest you to check [Let's encrypt, the free, automated and open certificate authority](#).
5. *YellowCube PHP SDK*: WooYellowCube depends on the [official PHP SDK](#). If you will install the plugin using the [suggested method](#) you don't need to do anything as it is already bundled in the official package.

You should also check the [WordPress](#) and [WooCommerce](#) requirements.

Installation

Getting the plugin

Upon installing [WordPress](#) and [WooCommerce](#) you'll proceed with WooYellowCube installation.

First you will need to download the latest version of the plugin from [its GitHub page](#). You will find the download packages in the releases section:



On the *Releases* page you will see the entire history of WooYellowCube releases and you will generally want to use the latest version (top on the list). Plugin package can be obtained through the links in the *Downloads* section.



Installation

Once you've successfully downloaded the package you will need to extract it into the site's directory structure. There are currently two ways to do that.

Manually

All Wordpress plugins are located under the `wp-content/plugins/` directory and the same is the case with the WooYellowCube. You should extract the package that you downloaded in the previous step and move its contents into the plugins directory.

If you've done this correctly you should have `wooyellowcube` inside the `wp-content/plugins` directory:



and file structure inside it that looks similar to this:

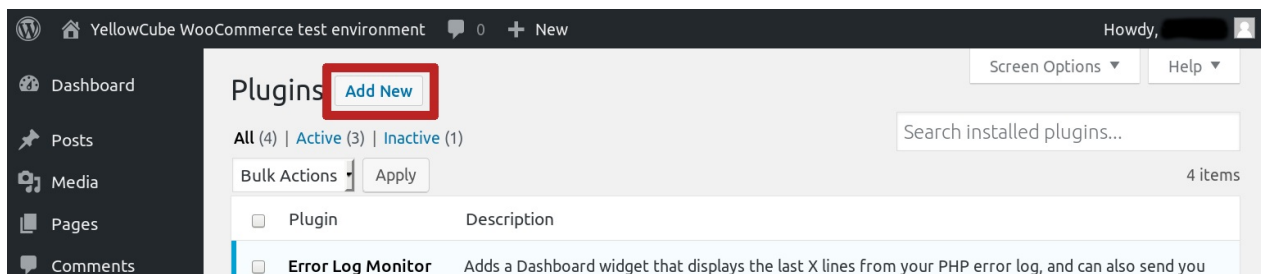


Depending on your hosting environment you will be required to use different tools to achieve this step (FTP, SSH, Git or other type of version control, ...). Consult your hosting provider if you are unsure about this.

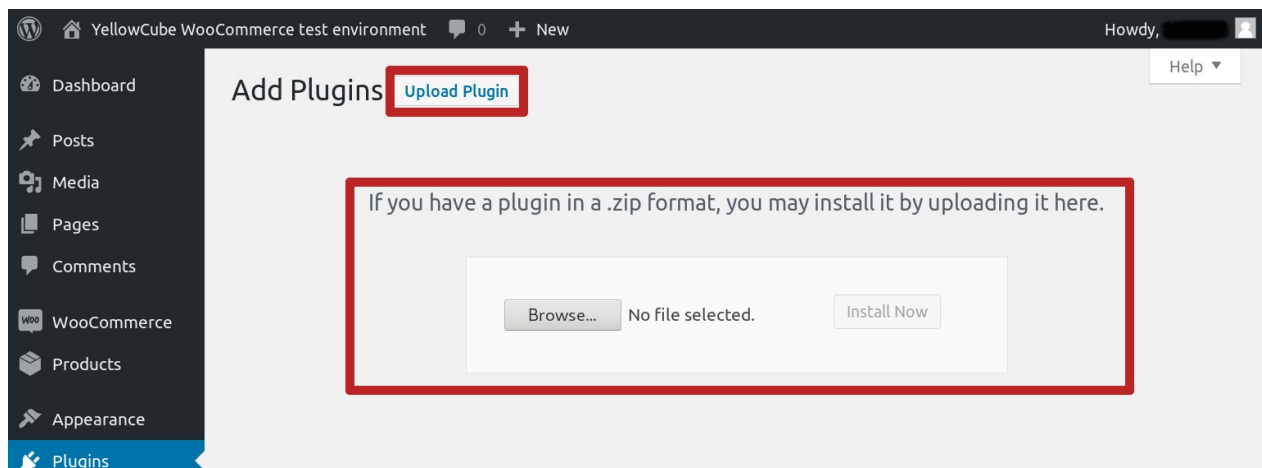
Using the admin interface

This method required web server to have write access to the plugin directory and `define('FS_METHOD', 'direct');` to be set in the `wp-config.php`.

You should navigate to *Plugins* section of the Wordpress administration pages and click on the *Add new* button:



This will take you to the plugin installation page where you are able to install plugins from the Wordpress plugin directory. Alternatively it is also possible to upload package that was downloaded in the first step. This can be achieved by pressing the *Upload* button and using the upload widget that shows up.



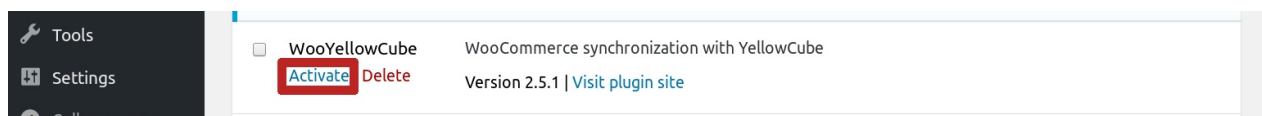
Please note that the upload widget only supports packages in `.zip` format so you need to make sure that you downloaded the correct package.

It is worth mentioning that the WooYellowCube plugin currently isn't available from the Wordpress plugin directory. We are aware that this would simplify the installation process significantly and are currently working on getting it included. We will update this document as soon as this is achieved.

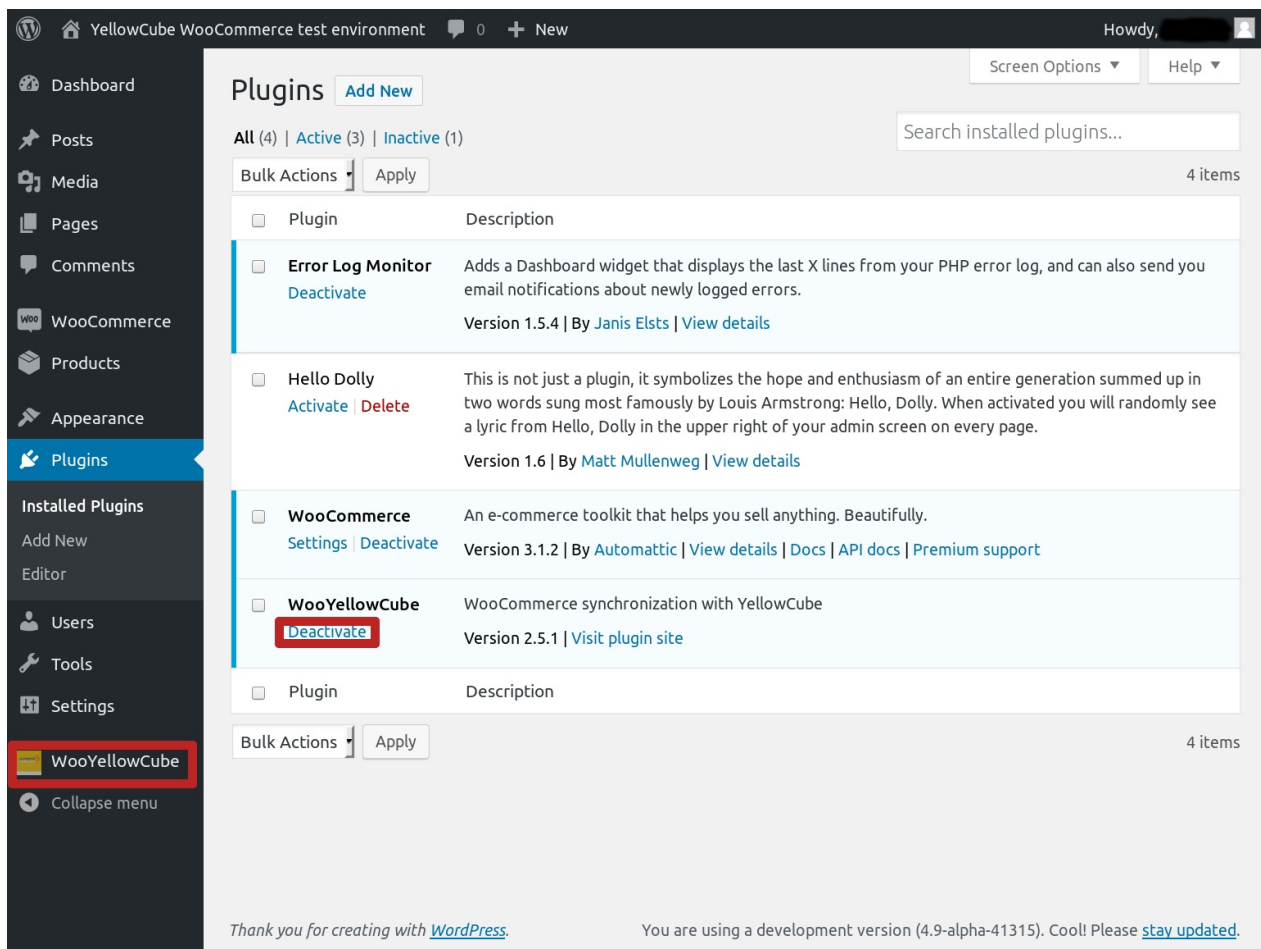
Activation

Once the plugin was successfully installed it will appear in the listing that is displayed in the *Plugins* section of the Wordpress administration pages.

Plugin can be activated by clicking on the *Activate* link next to it.



Once the plugin has been successfully activated the *Deactivate* link will appear next to it and the *WooYellowCube* section will become available in the administration menu.



Cron for background processes

The YellowCube plugin needs multiple recurring background processes to check for successful order processing and finally update the order with track & trace information.

Without any specific Cron setup, the YellowCube plugin will check for and trigger overdue background processes on every website request. This can randomly cause slow page requests.

You can avoid this additional overhead by setting up a dedicated System Cron. As this is hosting specific, check the documentation of your hosting provider how to setup Cron.

Disable the fallback cron

Edit `wp-config.php` and add the following line right BEFORE the line with `"wp-settings.php"`

```
define('DISABLE_WP_CRON', 'true');
```

As a result, all background updates will stop temporarily.

Setup the system cron

These are the endpoints to call the cron-jobs:

- Typically every 1..5 minute, checks for successful order / product submission if any is pending.
https://example.com/?cron_response=true
Changes order / product status from "Submitted" to "Confirmed" or "Error". (WAB, ART)
- Typically every 30 mins or hourly, checks for order shipping if any is pending (max 10 days).

https://example.com/?cron_hourly=true

Saves the track & trace number on order. (WAR)

- Typically daily, updates inventory stock information from YellowCube.
recommended at 06:00 after YellowCube exported the inventory at 04:00-04:30

https://example.com/?cron_daily=true

Replace "example.com" with your Site URL.

As an example, on a linux server with SSH terminal access you can typically setup Cron with crontab:

```
-$ crontab -l
# Every 5 minutes:
*/5 * * * * wget -q --delete-after https://example.com/?cron_response=true
# Every 30 minutes:
*/30 * * * * wget -q --delete-after https://example.com/?cron_hourly=true
# Every day at 06:00:
0 6 * * * wget -q --delete-after https://example.com/?cron_daily=true
```

Check the [Activity log](#) and make sure the background processes are called.

Configuration

Basic configuration

You will find the YellowCube configuration section under the *WooYellowCube* main menu item in the side administration toolbar.

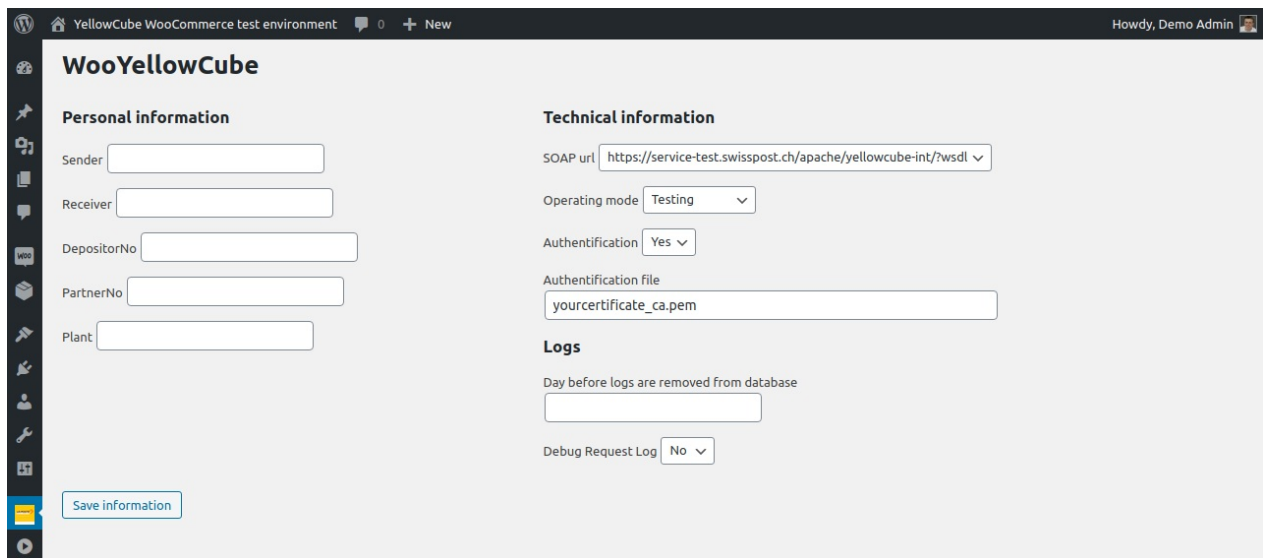
The fields below **Personal informations** need to be filled out with the information you have received by the YellowCube customer service. The field "Sender" is case sensitive.

The fields below **Technical information** need to be setup.

- **SOAP url:** URLs of SOAP services for different operating modes. <https://service.swisspost.ch/apache/yellowcube/?wsdl> should be used for production websites.
- **Operating mode:** Choose between *Testing and Production*
 - In Testing mode orders are processed virtually. Documents, labels etc. are printed out and scanned. This mode is for testing purposes.
 - Production mode needs to be set when the website is live and working.
 - Development mode is only intended to be used by developers. YellowCube users should never use this mode.
- **Authentication:** Should be enabled if you are using certificate file to authenticate with YellowCube.
- **Authentication file:** If the certificate authentication is used provide path to the certificate file on the server in this field (relative to the folder of the WooYellowCube extension).

Define how many days logs will be kept in the database in the **Logs** section. We suggest you start with 30 days and then reduce it if the amount of log items becomes an issue.

Note that you need to provide the IP address of your WooCommerce Shop to YellowCube for giving your shop access to YellowCube. This can also affect the configuration for any staging/development environments. Consult your dedicated service provider for more informations about that.



The screenshot shows the 'WooYellowCube' configuration page. It has a sidebar with various icons and a top bar with 'YellowCube WooCommerce test environment' and 'Howdy, Demo Admin'. The main content area is divided into two columns: 'Personal information' and 'Technical information'.

Personal information:

- Sender:
- Receiver:
- DepositorNo:
- PartnerNo:
- Plant:

At the bottom of this section is a 'Save information' button.

Technical information:

- SOAP url:
- Operating mode:
- Authentication:
- Authentication file:

Logs:

- Day before logs are removed from database:
- Debug Request Log:

Every time you save the settings form all states and caches are cleared. Background processes will re-run with the next trigger. This reduces issues when switching between Testing / Production and different accounts. However product registration states remain stale.

Limitation: Do NOT connect multiple shop instances to the same YellowCube account as WAR request (hourly cron containing confirming shippings with track & trace) can not be shared.

Production go live

Once you are ready for production use, change the operating mode settings accordingly:

The screenshot shows a configuration form with two main sections: **Personal information** and **Technical information**. On the left, there is a vertical sidebar with icons for various settings. The **Personal information** section includes input fields for Sender, Receiver, DepositorNo, and PartnerNo. The **Technical information** section includes a SOAP url dropdown (set to https://service.swisspost.ch/apache/yellowcube/?wsdl), an Operating mode dropdown (set to Production), an Authentication dropdown (set to Yes), and an Authentication file input field (set to yourcertificate_ca.pem).

Note that all previously registered articles need to be sent again to registration in production. There is no bulk operation yet to automate this process.

Limitation: If a previously successfully submitted product is not resent after switching the operating mode or account, order submissions containing such a product will fail.

Shipping

YellowCube relies on [shipping zones \(WooCommerce documentation\)](#) so you should configure them according to your business logic first. If unsure how to do that check the [relevant section of the WooCommerce documentation pages](#).

Once you have configured shipping zones and added the desired shipping methods to them you will be able to configure YellowCube specific configuration for each of them. You will be able to do that on the configuration page that can be found by navigating to *WooYellowCube -> Shipping*:

The screenshot shows the **Shipping management** interface. It has a header bar with the environment name 'YellowCube WooCommerce test environment' and a user profile 'Howdy, Demo Admin'. The main section is titled **Zones shipping** with a subtitle 'Attribute YellowCube shipping methods to zone shipping methods.' Below this, there is a section for **Switzerland** containing a table of shipping methods. The table has columns for Name, Status, BasicShippingServices, AdditionalShippingServices, and Action. Three methods are listed: PostPak PRI Flat, PostPak ECO Flat, and Pickup. Each method has a Status dropdown set to 'Activated', a BasicShippingServices dropdown set to 'PRI', 'ECO', and 'PICKUP' respectively, and an AdditionalShippingServices dropdown set to 'None'. Each row has an 'Update' button. Below the Switzerland section, there is a section titled **Locations not covered by your other zones** containing a table with one method: 'Flat rate test', which is also 'Activated' and has 'ECO' as the BasicShippingServices and 'None' as the AdditionalShippingServices, with an 'Update' button.

The page lists all shipping methods (grouped by zones) and allows you to configure YellowCube for each of them separately. YellowCube can be enabled by selecting *Activated* in the *Status* dropdown for the relevant shipping zone/method.

Orders still can be manually sent to YellowCube on each [order detail page](#) even if a matched zone is deactivated for YellowCube.

The other two dropdowns (*BasicShippingServices* and *AdditionalShippingServices*) allow you to select the YellowCube product that should be used. You will find more detailed information about those in the YellowCube Interface Manual.

Activity Log

Check the [activity log](#) to see the activity on your website.

Compatibility

Tested with

- *WPML* for a multilingual shop. Still known to causes poor performance.
- *Error Log Monitor* to better identify runtime errors.

Known incompatibilities

Do not use any other multilingual modules except WPML as they have known issues.

Do not use any module that modifies shipping rules.

Update

Please follow this guideline to update your Plugin in an existing installation.

Getting the latest version of plugin

Download the latest version of the plugin, see [installation - Getting the plugin](#).

Verify your installation

Before updating your installation, we recommend to double check that you previously followed the recommendations for the [installation](#). Some recommendations have been updated with recent releases.

Save your certificate

Make sure to backup your certificate in a safe place before you perform any update. Your current certificate may be in the modules' folder and may be lost if the folder is replaced in full during the update. You can find your certificate path in the [Configuration](#) screen labelled as "Authentication file". The certificate file typically has a file ending `.pem`. Backup your certificate from the given path.

If you accidentally lost the certificate file, maybe by overwriting, request a new certificate via [YellowCube Support](#).

Method A: Manual replacement

This method is recommended for professionals and works if you have access to the filesystem of your server with tools like ssh, FTP.

After [saving the certificate file](#), delete the plugin folder `wooyellowcube` inside the `wp-content/plugins` directory. Do not disable the plugin. Double check that there is no other outdated "backup" copy of the plugin in a renamed folder as this can lead to strange bugs. If you want to keep any old version of the plugin, move it out of the WordPress installation folder.

Extract the downloaded .zip file, into a new folder `wooyellowcube`. Make sure there is no additional nested folder and `index.php` is directly contained. Move the new folder, containing the new version, to the installations' `wp-content/plugins` directory. Recover the certificate in the modules' folder with the same filename.

Login to WordPress and check the YellowCube logs for the database update message, [see blow](#).

Method B: Using the admin interface

This method works if you are a WordPress administrator and the web server has write permissions, even if you have no direct server access with tools like ssh or FTP.

After [saving the certificate file](#), you will package the downloaded .zip with your certificate file to make it work with the admin interface installer.

Rename the file `wooyellowcube-master.zip` to `wooyellowcube.zip`.

Open the file `wooyellowcube.zip` with your favorite archive program and rename the containing folder `wooyellowcube-master` to `wooyellowcube`. Enter the folder `wooyellowcube` in the zip repository and drag-and-drop the previously [saved certificate file](#) from the file explorer. Save the archive and double check that the folder is renamed and the certificate is contained.

Go to the "Plugins" area in WordPress.

The screenshot shows the WordPress dashboard for a 'YellowCube WooCommerce test environment'. The 'Plugins' menu is active, displaying a list of installed plugins. The 'WooYellowCube' plugin is selected, and its 'Deactivate' button is highlighted with a red box. The plugin description states it is for 'WooCommerce synchronization with YellowCube' and is version 2.5.5. Other visible plugins include 'WooCommerce' and 'WooCommerce Admin'.

Then identify the plugin "WooYellowCube" and click "Deactivate" and next "Delete" and confirm.

This is a close-up of the 'WooYellowCube' plugin entry. The 'Delete' button is highlighted with a red box, indicating the next step in the process.

On the Plugin header click "Add new" and then "Upload Plugin".

The first screenshot shows the 'Add New' button in the 'Plugins' header highlighted with a red box. The second screenshot shows the 'Upload Plugin' button in the 'Add Plugins' section highlighted with a red box. The 'Add Plugins' page includes a text prompt: 'If you have a plugin in a .zip format, you may install it by uploading it here.' and a file upload area with a 'Choose file' button and an 'Install Now' button.

Then "Choose file" and browse for the prepared `wooyellowcube.zip`, select and click "Open" (might vary depending on your system) and then "Install Now".

This screenshot shows the 'Installing Plugin from uploaded file: wooyellowcube.zip' screen. The progress bar indicates 'Plugin installed successfully.' The 'Activate Plugin' button is highlighted with a red box, and a 'Return to Plugin Installer' link is also visible.

Finally click "Activate Plugin" to finish the installation.

Login to WordPress and check the YellowCube logs for the database update message, [see blow](#).

Always test after updates

Make sure to test the connection after any update.

We recommend you to go to the "WooYellowCube" [configuration page](#) and hit "Save information".

Then check the [stock table](#) with "Force to refresh inventory" and click "Execute". If YellowCube already has your goods on stock, the list will be provided.

Additionally, we highly recommend you to check [updating a product](#) and [place an order](#) to make sure everything works as expected.

If there is an unexpected error message, first double check the certificate path and filename matches the configuration.

New in latest 2.5.5

The version 2.5.5 contains many bugfixes, stock management improvements and compatibility with WordPress 5.0+ and PHP 7.2.

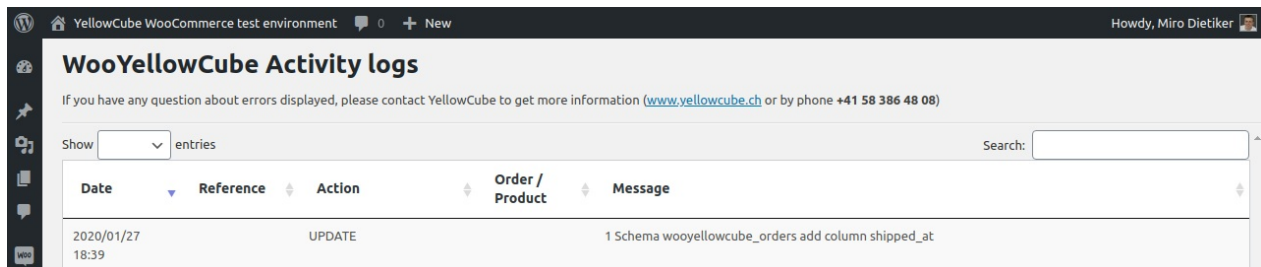
Recommended to setup dedicated Cron

The plugin has now improved support for setting up a dedicated cron for stability and better performance. We highly recommend setting it up. See [installation - Cron for background processes](#).

Automatic database update

Version 2.5.5 will perform a database schema update on the first page request to your WordPress administration.

A successful schema update will be reported in the [activity log](#) as "1 Schema wooyellowcube_orders add column shipped_at".



Date	Reference	Action	Order / Product	Message
2020/01/27 18:39		UPDATE		1 Schema wooyellowcube_orders add column shipped_at

Store administration guide

This section covers most important information related to the management of the WooCommerce store. This includes product and order creation and management.

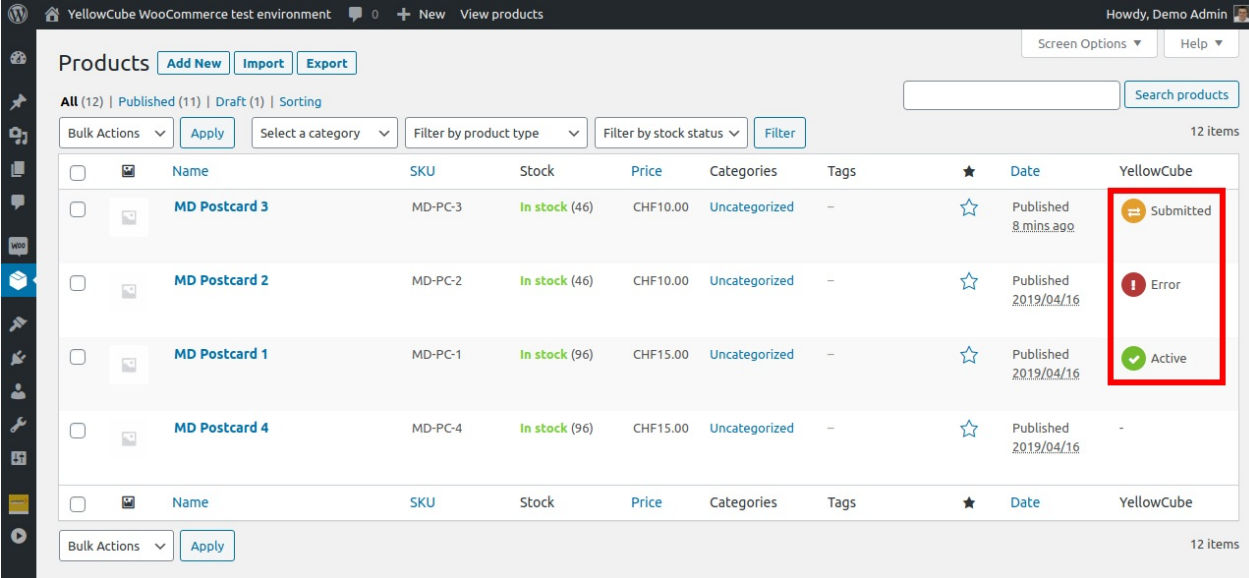
YellowCube is just a tiny part of the entire process. We suggest store admins to study the official WooCommerce documentation that covers [product](#) and [order](#) management.

Product management

A product that is managed by YellowCube needs to be registered first.

Product list

The registration status of each products can be seen on the product listing page:



	Name	SKU	Stock	Price	Categories	Tags	★	Date	YellowCube
<input type="checkbox"/>	MD Postcard 3	MD-PC-3	In stock (46)	CHF10.00	Uncategorized	-	☆	Published 8 mins ago	Submitted
<input type="checkbox"/>	MD Postcard 2	MD-PC-2	In stock (46)	CHF10.00	Uncategorized	-	☆	Published 2019/04/16	Error
<input type="checkbox"/>	MD Postcard 1	MD-PC-1	In stock (96)	CHF15.00	Uncategorized	-	☆	Published 2019/04/16	Active
<input type="checkbox"/>	MD Postcard 4	MD-PC-4	In stock (96)	CHF15.00	Uncategorized	-	☆	Published 2019/04/16	-

The possible values in the YellowCube column are:

- -: Product is not connected to YellowCube.
- *Submitted*: An new product definition or an update was sent to YellowCube and we are waiting for confirmation. New products are not yet connected to YellowCube.
- *Active*: YellowCube confirmed that it received and processed the product information. Everything seems to be OK.
- *Error*: YellowCube was unable to process the product information. Check the log on the product edit page to learn more.

Edit product details

Detailed information about YellowCube communication about the product is available at the bottom of the product edit page.

In order to register a product at YellowCube successfully it needs to meet the following requirements:

- *SKU*: required field, used as primary product identifier. Changes are not supported after registration at YellowCube. You will find it under *Product data* -> *Inventory*.

Edit product [Add New](#)

MD Postcard 1

Permalink: <http://ycwordpress.staging.office.md-systems.ch/product/md-postcard-1/> [Edit](#)

[Add Media](#) Visual Text

Word count: 0 Last edited by Demo Admin on August 20, 2019 at 1:04 pm

Product data — Simple product Virtual: ☐ Downloadable: ☐

- General**
- Inventory**
 - SKU: MD-PC-1
 - Manage stock? ☒ Enable stock management at product level
 - Stock quantity: 96
 - Allow backorders? Do not allow
 - Low stock threshold: 2
 - Sold individually: ☐ Enable this to only allow one of this item to be bought in a single order
- Shipping**
- Linked Products**
- Attributes**
- Advanced**
- Get more options**

Publish

Status: **Published** [Edit](#)

Visibility: **Public** [Edit](#)

Published on: Apr 16, 2019 at 08:41 [Edit](#)

Catalog visibility: **Shop and search results** [Edit](#)

[Copy to a new draft](#) [Move to Trash](#) [Update](#)

Product categories

All categories Most Used

☒ Uncategorized

[+ Add new category](#)

Product tags

[Add](#)

Separate tags with commas

[Choose from the most used tags](#)

- **Weight:** in kg, required field. You will find it under *Product data* -> *Shipping*.
- **Dimensions:** in cm, not strictly required. However, it is recommended to provide this information.
- **Manage stock:** required to enable for products managed with YellowCube.

Word count: 0 Last edited by Demo Admin on November 20, 2019 at 1:57 pm

Product data — Simple product Virtual: ☐ Downloadable: ☐

- General**
- Inventory**
- Shipping**
 - Weight (kg): 0.1
 - Dimensions (cm): 10 15 1
 - Shipping class: No shipping class
- Linked Products**
- Attributes**
- Advanced**
- Get more options**

Product image

[Set product image](#)

Product gallery

In order to trigger the registration at YellowCube you need to click *Send product to YellowCube*. Make sure to save any article change prior to click this button as changes are lost.

For later product updates the button is later named *Update product to YellowCube*.

Product registration and update

In the left column we can see the general status of the product.

Log of all the communication about the product can be seen in the right column. If there were any problems you will be able to see the error messages here.

Disconnect from YellowCube

A product can be disconnected from YellowCube by clicking the button *Remove the link with YellowCube*. The button is only visible for properly registered products.

Disconnected products will not be sent to YellowCube as order items if they are part of an order. See also [Order management - Order completion](#)

If you later reconnect the product again, it will match the previously created product at YellowCube again based on the SKU. Creating a new record requires manual deletion by the [YellowCube team](#).

Advanced: Using EAN

You can pass EAN numbers to YellowCube by attaching a custom attribute named *EAN* to the product.

EAN codes typically contain 8 or 13 characters. YellowCube will use them to better identify labelled the products.

Advanced: Lot management

In order to deal with expiring products, you can enable lot management by choosing *Lot management is enabled* and submitting a product update.

You can't deactivate lot management once it has been enabled. Even after disconnecting the product, it needs manual delete at YellowCube to re-add it without lot management again. Before enabling lot management you need to consider all implications of such action. If you are unsure about that please contact Swiss Post support for more information.

Unsupported: Multiple articles with the same SKU

WooYellowCube doesn't support using the same SKU in multiple articles or variations. As a result, each product would have an independent stock counter and a random one would be picked for the update. This can be a limitation with certain multilingual approaches.

Troubleshooting Product deletion and trash

If you ever create a duplicated product with the same SKU and delete it again, reconnect the original product again by choosing "Update to YellowCube" on the product edit page. Try to avoid duplicate SKUs whenever possible. Either edit the product and immediately change the SKU when duplicating or later delete it for real.

It is highly recommended not to delete unique products (with distinct SKU) as it can cause errors and inconsistencies when processing orders that contained these products. You can send the products to the product trash and keep them.

Order management

An order is relevant for YellowCube if the determined shipping zone is enabled for YellowCube.

When relevant orders are created they will be sent to YellowCube as soon as they reach the *Processing* status.

A well integrated online payment method (e.g. credit cards) will update the status either immediately or with a short delay after clearance (based on a callback).

If the selected payment method requires manual intervention, the order remains in status *On hold* until the store admin changes it to *Processing* manually upon payment.

Order list

The status of each orders can be seen on the order listing page:

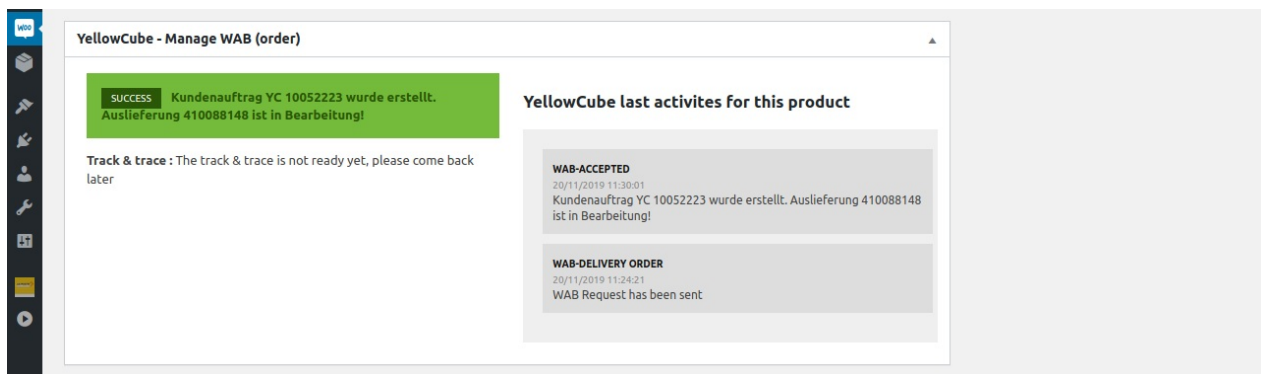
Order	Date	Status	Total	YellowCube
#185 Test Client	36 mins ago	Failed	CHF15.00	Error
#184 Test Client	37 mins ago	On hold	CHF13.00	-
#183 Test Client	45 mins ago	On hold	CHF13.00	Submitted
#182 Test Client	45 mins ago	Processing	CHF13.00	Confirmed
#181 Test Client	3 hours ago	Processing	CHF13.00	Confirmed
#180 Test Client	3 hours ago	Completed	CHF13.00	Confirmed
#179 Test Client	3 hours ago	Processing	CHF13.00	Confirmed

The possible values in the YellowCube column are:

- -: Order has not been sent to YellowCube at all. This could mean that this specific order shouldn't be processed by YellowCube (sent to a shipping zone that is not handled by YellowCube) or that there are still conditions to be met to do so (usually awaiting payment).
- Submitted: Order has been sent to YellowCube but it hasn't been processed on their side yet.
- Confirmed: Order was dispatched from the YellowCube facility and the product is on its way to the customer. Tracking code is provided in the logs if applicable.
- Error: A problem occurred while trying to process the order on the YellowCube side. Check the logs on the order edit page for more info.

Edit Order for details

Detailed information about YellowCube communication about each order is available at the bottom of the order edit page.



The status of the order and the tracking information (if available) are displayed on the left.

Log of all the communication about the order can be seen in the right column. If there were any problems you will be able to see the error messages here.

Limitation: If an order matched by a YellowCube enabled zone contains no single article effectively managed by YellowCube, the submission fails with error "*The content of element 'ns1:OrderPositions' is not complete.*"

Thus if choose YellowCube with WooCommerce, it is highly recommended to register all products at YellowCube.

Stock reservation on submission

Any order submission will reserve its articles and deduct each articles stock counter immediately.

Proper order cancellation

An order can stay stuck in status *Pending payment* or maybe *Failed* if a customer cancels on the payment page. Customers choosing manual payment methods in status *On hold* might change their mind about the order. See also the official [status definition \(WooCommerce documentation\)](#).

In order to free the reserved stock on cancellation, WooCommerce requires a store admin to put the order to status *Cancelled*.

Permanently deleting an order with reserved stock via trash will not restore the stock counter. This is a WooCommerce limitation.

Stop order delivery

Once an order is submitted to YellowCube, you can not stop its delivery in WooCommerce.

A store admin can try to stop delivery by calling [calling YellowCube Support](#) as long as the delivery was no yet started.

Additionally, the store admin will need to manually set the status to *Cancelled* to free its stock again.

Returns

Please note that the WooYellowCube currently doesn't handle order returns automatically. If a return happens the affected order will remain unchanged, the payment refund and any other relevant steps need to be made manually by the store admin. The stock increase will be visible in the stock table the next day after the return.

Refer to the [WooCommerce refund documentation](#) . It is recommended to process returns with refunds based on items and choose "Restock refunded items". As a result the YellowCube stock list will report a difference until it reports the return the next day. Alternatively if you wait with manual refund handing until the day after the return, the YellowCube stock will first report a difference until you report the item as refunded.

Order Completion

As soon as an order track & trace information is received, the order status is set to Completed. An order can receive multiple track & trace numbers.

YellowCube does NOT support order splitting, it is either processed in full or not.

WooCommerce does NOT support multiple stores or a UI for order splitting.

Limitation: As a result, orders reach the completed state even if some of their products are not registered / shipped by YellowCube.

Backed order creation

Backend order creation is supported and such orders can be force submitted to YellowCube. Force submitting is required since the shipping zones are never matched. It might be still easier to use the public store interface to create an order on behalf of the customer.

Advanced: Manual order submission

Some store admins like to determine on a per order basis if they handle it on their own or pass it to YellowCube. If YellowCube is disabled for a matched shipping zone, a store admin can still manually force submit it.

Troubleshoot order submission errors

If the order triggers a submission error, make sure first that all products are properly registered at YellowCube by resubmitting them with the button *Update product to YellowCube*. Then try submitting the order again.

If a product has never been sent to YellowCube before, it is skipped on order submission and expected to be manually processed. In order to submit such a product to YellowCube, you need to create a new order after product registration.

Stock management

In the **Stock management** you will find the effective YellowCube product stock for each *Product name (SKU)* and compare it with the local WooCommerce stock counter.

To understand potential stock counter inconsistencies, we also display the pending article count.



SKU	Shop product	Shop stock	Shop pending	YellowCube stock	YellowCube date	Stock similarity	Details
MA-PG-PC-NRBC-1	KB Product 1	85	40	125	20/11/2019 04:01	Same stock	View

Action on selected products

Send ART profile

Execute

This contains all articles in received orders that are not yet processed by YellowCube.

Columns are:

- *SKU*: The unique product identifier
- *Shop product*: A link to the connected WooCommerce product
- *Shop stock*: Available stock as of the shop counter.
- *Shop pending*: Orders that are not yet sent to YellowCube or outstanding confirmation.
- *YellowCube stock*: The effective YellowCube stock at date.
- *YellowCube date*: The date of the last stock report.
- *Stock similarity*: * Different stock* if it doesn't match.
- *Details*: Access to optional lot management details.

Stock count math

YellowCube exports each night at about 04:00 the current stock information. The daily cron then updates the stock table with the information. Even on force refresh, the stock information is not more up-to-date.

On the other hand, WooCommerce decreases the stock as soon as an order is received.

Thus the stock is considered the *Same stock* if

Shop stock + Shop pending == YellowCube stock

Accordingly an update of *Shop stock* will deduct *Shop pending* from the *YellowCube stock*

Thus WooCommerce [does not fit for splitting orders and managing stock across multiple stores](#) including additional local product storage.

Stock actions

Here you also can take action for selected products:

- *Send ART profile* Will send basic information (name, SKU, size, weight, ...) about the product to YellowCube. This generally happens automatically upon product creation and update so it should only be used if something went wrong.
- *Update WooCommerce Stock with YellowCube*: If you are sure that the stock information provided by YellowCube is correct you can use this action to override the WooCommerce stock information.
- *Force to refresh inventory* This will reload the stock table data from YellowCube based on the last inventory export.

Activity Log

Check the **activity log** to see the activity on your website. A log message is automatically created when errors occur, stock inventory is updated etc. You will also find the tracking numbers for orders that were already processed and shipped.

Date	Reference	Action	Order / Product	Message
2019/11/20 14:10		WAB-ACCEPTED	#183	Kundenauftrag YC 10052226 wurde erstellt. Auslieferung 410088151 ist in Bearbeitung!
2019/11/20 14:05	102115	ART-ACCEPTED	MD-PC-3	Der Artikel wurde angelegt bzw. erweitert.
2019/11/20 14:04	102117	WAB-DELIVERY ORDER	#183	WAB Request has been sent
2019/11/20 13:57	102115	ART-UPDATE	MD-PC-3	File received
2019/11/20 13:53	102114	ART-UPDATE	MD-PC-3	File received
2019/11/20 13:50	102111	ART-ACCEPTED	MD-PC-2	Der Artikel wurde angelegt bzw. erweitert.
2019/11/20 13:50	102113	ART-ACCEPTED	MD-PC-3	Der Artikel wurde angelegt bzw. erweitert.
2019/11/20 13:47	102113	ART-INSERT	MD-PC-3	File received

Each activity log that is related to an order or product will output a link for quick access.

Debug Request Log for Support

To provide more efficient support and help fixing problems, the plugin provides a [Debug Request Log](#) to allow deeper plugin activity insights.

There will be additional items with Action such as:

- **SERVICE:** Any request to a YellowCube service with contextual information.
- **CRON-RESPONSES:** Any responses cron trigger, typically every 1..5 minutes.
- **CRON-HOURLY:** Any hourly cron trigger.
- **CRON-DAILY:** Any daily cron trigger.