

Partner Order API

Version 1.0 · Reference for dropshipping partners

This document replaces the previous Turor OMS package (`POST /restApi/v1/dropShippingOrder` , `GET /restApi/v1/dropShippingOrderItem/{id}`). Those endpoints and their HTTP Basic credentials are **deprecated** and must no longer be used. See **Appendix A** at the end of this page for the field-by-field mapping.

Do you need this API?

Most partners do not. If you run WooCommerce or Shopify, use our plugin instead — see WooCommerce or Shopify. Build against this API only when you have a custom or in-house system.

1. Environments

Environment	Base URL
Production	<code>https://cosmeticwholesale.eu/api/v1</code>
Staging (integration testing)	<code>https://cosmeticwholesale1.bookgurusapi.com/api/v1</code>

All examples in this document use the production base URL. Tokens are **not** shared between environments — ask your account manager for a staging token if you want to test first.

2. Authentication

Every request carries your **account token** as a bearer token:

```
Authorization: Bearer YOUR_ACCOUNT_TOKEN
```

This is the same token you use for the product catalog feed (`GET /api/v1/feeds/catalog.csv`). You can find it on the **Account** page of your Cosmetic Wholesale login, together with your feed URL.

Notes:

- The token identifies your account. Never expose it in client-side code or a public repository.
- Tokens can be revoked and reissued from your account at any time. A revoked token returns `401`.
- No username/password is used. The old HTTP Basic credentials are no longer valid for order submission.
- `X-Feed-Token: YOUR_ACCOUNT_TOKEN` is accepted as an alternative header.

3. Prerequisites

An order is only accepted when all of the following are true:

1. Your account is **approved** and on the **dropshipping** channel.

2. Your dropshipping **subscription is active** — otherwise every request returns **403** `subscription_required`.
3. Your **wallet has enough balance** to cover the order — otherwise the order is created and held (**402**, see [section 6](#)).
4. Every **ean** in the order exists in your catalog feed and is in stock.

Product titles, images and prices are **not** sent by you — they come from our catalog. You only send the EAN and the quantity.

4. Create an order

```
POST /dropshipping/orders
```

Request headers

```
Authorization: Bearer YOUR_ACCOUNT_TOKEN
Content-Type: application/json
```

Request body

```
{
  "externalRef": "ORD123456",
  "payment": "WALLET",
  "shipping": {
    "firstName": "Name",
    "lastName": "LastName",
    "company": "TestCompany",
    "address1": "Main Street 123",
    "address2": "",
    "postcode": "1072 HM",
    "city": "Amsterdam",
    "country": "NL",
    "phone": "",
    "email": "customer@example.com"
  },
  "lines": [
    { "ean": "6291107455365", "quantity": 2 },
    { "ean": "3607348816552", "quantity": 3 }
  ]
}
```

Fields

Field	Type	Required	Notes
<code>externalRef</code>	string	yes	Your own order number. Must be unique within your account . This is the idempotency key — see section 5.
<code>payment</code>	string	no	Only <code>"WALLET"</code> is supported. Defaults to <code>"WALLET"</code> .
<code>lines[].ean</code>	string	yes	EAN as published in your catalog feed.
<code>lines[].quantity</code>	integer	yes	Must be ≥ 1 .
<code>shipping.firstName</code>	string	recommended	
<code>shipping.lastName</code>	string	recommended	
<code>shipping.address1</code>	string	yes	
<code>shipping.address2</code>	string	no	
<code>shipping.postcode</code>	string	yes	
<code>shipping.city</code>	string	yes	
<code>shipping.country</code>	string	yes	ISO 3166-1 alpha-2, e.g. <code>NL</code> , <code>BE</code> , <code>DE</code> . Must be a country we ship to.
<code>shipping.company</code>	string	no	
<code>shipping.state</code>	string	no	
<code>shipping.phone</code>	string	no	Recommended — some carriers require it.
<code>shipping.email</code>	string	no	Used for carrier notifications where supported.

`snake_case` aliases are accepted for compatibility: `external_ref`, `first_name`, `last_name`, `address_1`, `address_2`, and `qty` for `quantity`.

Response

```
{
  "id": "8f1c0b2e-6a3d-4f21-9b7e-0c5a4d2e1f33",
  "externalRef": "ORD123456",
  "omsReference": "ds:usr_2f9a1c:ORD123456",
  "createdVia": "dropshipping-api",
  "status": "PAID",
  "walletDebited": true,
  "omsPushStatus": "pending",
  "omsOrderItemIds": [],
  "tracking": []
}
```

Field	Meaning
<code>id</code>	Cosmetic Wholesale order id (UUID). Store this.
<code>externalRef</code>	Echo of your order number.
<code>omsReference</code>	Internal warehouse reference, <code>ds:{accountId}:{externalRef}</code> .
<code>status</code>	See section 7.
<code>walletDebited</code>	<code>true</code> once the order has been paid from your wallet.
<code>omsPushStatus</code>	Progress of the hand-off to our warehouse: <code>pending</code> , <code>pushed</code> , <code>retry</code> , <code>failed</code> , <code>skipped</code> , or <code>null</code> .
<code>omsOrderItemIds</code>	Warehouse line ids, filled in after the hand-off completes. Empty on the first response — poll with <code>GET</code> .
<code>tracking</code>	Empty until the parcel ships. See section 8.

HTTP status codes

Code	Meaning	What your system should do
<code>201</code>	Order created and paid from the wallet.	Mark as sent.
<code>200</code>	This <code>externalRef</code> already exists. The body is the current state of that order.	Nothing — this is a safe retry.
<code>402</code>	Order created but held : wallet balance is insufficient.	Do not cancel. See section 6.
<code>400</code>	Validation error.	Fix the payload; do not retry unchanged.
<code>401</code>	Missing, invalid or revoked token.	Check credentials.
<code>403</code>	No active dropshipping subscription.	Contact us.
<code>404</code>	(GET only) Order not found on your account.	
<code>409</code>	Insufficient stock for one or more lines.	Retry later or reduce the quantity.

5. Idempotency

`externalRef` is the idempotency key. Posting the same `externalRef` twice never creates a second order and never debits your wallet twice:

- If the first attempt succeeded, the second returns `200` with the existing order.
- If the first attempt was held for insufficient balance, the second attempt retries the wallet payment and returns `201 / 200` once the balance covers it.

This makes the endpoint safe to retry on timeouts and network errors. Always retry with the same `externalRef` — never generate a new one for the same customer order.

6. Insufficient balance: the `402` HOLD contract

This behaviour has no equivalent in the old OMS API and is the most common integration mistake.

When your wallet cannot cover the order:

HTTP/1.1 402 Payment Required

```
{
  "id": "8f1c0b2e-6a3d-4f21-9b7e-0c5a4d2e1f33",
  "externalRef": "ORD123456",
  "status": "HOLD",
  "reason": "INSUFFICIENT_WALLET",
  "walletDebited": false,
  "omsPushStatus": null,
  "omsOrderItemIds": [],
  "tracking": []
}
```

What this means:

- The order **exists** in our system and is reserved. It is not rejected and not deleted.
- Nothing has been debited.
- Nothing has been sent to the warehouse yet.

What your system must do:

- **Do not cancel or delete the order** on your side. Keep it in a pending/processing state.
- Top up your wallet.
- Re-POST the **same externalRef**, or simply wait: held orders are retried automatically when the balance becomes sufficient.
- Poll **GET** until **walletDebited** is **true** and **status** is no longer **HOLD**.

Cancelling on **402** is what breaks reconciliation — the order stays reserved with us while your system believes it failed.

7. Read an order

```
GET /dropshipping/orders/{id}
GET /dropshipping/orders?externalRef=ORD123456
```

{id} accepts either the Cosmetic Wholesale **id** (UUID) or your own **externalRef**. Both forms return the same body as the create call. Orders are scoped to your token — you can only read your own orders.

```
GET https://cosmeticwholesale.eu/api/v1/dropshipping/orders?externalRef=ORD123456
Authorization: Bearer YOUR_ACCOUNT_TOKEN
```

Order status values

status	Meaning
HOLD	Held for insufficient wallet balance (see section 6).
PENDING_PAYMENT	Wallet payment in progress.
ON_HOLD	Awaiting shipping/VAT processing.
PAID	Paid from wallet.
PROCESSING	Being prepared in the warehouse.
SHIPPED	Handed to the carrier — tracking is populated.
COMPLETED	Delivered / closed.
CANCELLED , REFUNDED	Terminal states.

8. Tracking

Tracking is returned on the same **GET** , as an array (an order can ship in several parcels):

```
{
  "id": "8f1c0b2e-6a3d-4f21-9b7e-0c5a4d2e1f33",
  "externalRef": "ORD123456",
  "status": "SHIPPED",
  "walletDebited": true,
  "omsPushStatus": "pushed",
  "omsOrderItemIds": ["418224", "418225", "418226"],
  "tracking": [
    {
      "provider": "postnl-3s",
      "trackingNumber": "3SXXX000000000",
      "dateShipped": "2026-05-06T17:39:20.000Z"
    }
  ]
}
```

Poll no more than once per hour per order. Do not poll orders that are already **COMPLETED** , **CANCELLED** or **REFUNDED** .

9. Error codes

Errors return `{ "error": "<message>", "code": "<code>" }` .

code	HTTP	Cause
external_ref_required	400	externalRef missing or empty.
lines_required	400	lines missing or empty.
ean_required	400	A line has no ean .
invalid_qty	400	quantity is not a positive integer.
unknown_ean	400	The EAN is not in your catalog.
forbidden_ean	400	The EAN cannot be ordered through this endpoint.
shipping_required	400	shipping object missing.
shipping_incomplete	400	address1 , city , postcode or country missing.
shipping_country_required	400	Country empty or not in our shipping zones.
payment_wallet_only	400	payment was set to something other than WALLET .
invalid_json	400	Body is not valid JSON.
—	401	Token missing, invalid or revoked.
subscription_required	403	No active dropshipping subscription.
not_found	404	No such order on your account.
insufficient_stock	409	Not enough stock for one or more lines.

10. Examples

cURL — create

```
curl -X POST https://cosmeticwholesale.eu/api/v1/dropshipping/orders \
-H "Authorization: Bearer YOUR_ACCOUNT_TOKEN" \
-H "Content-Type: application/json" \
-d '{
  "externalRef": "ORD123456",
  "shipping": {
    "firstName": "Name",
    "lastName": "LastName",
    "address1": "Main Street 123",
    "postcode": "1072 HM",
    "city": "Amsterdam",
    "country": "NL"
  },
  "lines": [
    { "ean": "6291107455365", "quantity": 2 }
  ]
}'
```

cURL — read

```
curl "https://cosmeticwholesale.eu/api/v1/dropshipping/orders?externalRef=ORD123456" \
-H "Authorization: Bearer YOUR_ACCOUNT_TOKEN"
```

PHP — create

```
<?php

$token = 'YOUR_ACCOUNT_TOKEN';
$url   = 'https://cosmeticwholesale.eu/api/v1/dropshipping/orders';

$payload = [
    'externalRef' => 'ORD123456',
    'shipping' => [
        'firstName' => 'Name',
        'lastName'  => 'LastName',
        'address1'  => 'Main Street 123',
        'postcode'  => '1072 HM',
        'city'      => 'Amsterdam',
        'country'   => 'NL',
    ],
    'lines' => [
        ['ean' => '6291107455365', 'quantity' => 2],
    ],
];

$ch = curl_init($url);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, [
    'Authorization: Bearer ' . $token,
    'Content-Type: application/json',
]);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($payload));

$response = curl_exec($ch);
$status   = curl_getinfo($ch, CURLINFO_HTTP_CODE);
curl_close($ch);

$order = json_decode($response, true);

if ($status === 402) {
    // Held for insufficient wallet balance.
    // Keep your order open and retry with the SAME externalRef after topping up.
} elseif ($status === 201 || $status === 200) {
    // Accepted. Store $order['id'] and poll for tracking.
}

?>
```

PHP — read

```
<?php

$token          = 'YOUR_ACCOUNT_TOKEN';
$externalRef    = 'ORD123456';
$url = 'https://cosmeticwholesale.eu/api/v1/dropshipping/orders/' . rawurlencode($externalRef);

$ch = curl_init($url);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, ['Authorization: Bearer ' . $token]);
$response = curl_exec($ch);
curl_close($ch);

echo $response;
?>
```

Appendix A — Migrating from the Turor OMS endpoints

The old package pointed at `https://omsback.turor.org/restApi/v1` with HTTP Basic credentials. Those credentials belonged to a shared system account and bypassed the wallet, stock and subscription layer. They are no longer issued and existing ones will be withdrawn.

Endpoints

Old	New
POST /restApi/v1/dropShippingOrder	POST /api/v1/dropshipping/orders
GET /restApi/v1/dropShippingOrderItem/{id}	GET /api/v1/dropshipping/orders/{id} (order level, not line level)
HTTP Basic username / password	Authorization: Bearer <account token>

Request fields

Old	New
<code>order_number</code>	<code>externalRef</code>
<code>line_items[]</code>	<code>lines[]</code>
<code>line_items[].ean</code>	<code>lines[].ean</code>
<code>line_items[].quantity</code>	<code>lines[].quantity</code>
<code>line_items[].name</code>	<i>removed</i> — taken from our catalog
<code>line_items[].image_src</code>	<i>removed</i> — taken from our catalog
<code>shipping.first_name</code> / <code>last_name</code>	<code>shipping.firstName</code> / <code>lastName</code> (snake_case still accepted)
<code>shipping.address_1</code> / <code>address_2</code>	<code>shipping.address1</code> / <code>address2</code> (snake_case still accepted)
<code>shipping.country</code> , <code>city</code> , <code>postcode</code> , <code>company</code> , <code>phone</code>	unchanged
—	<code>payment: "WALLET"</code>

Response fields

Old	New
<code>orderCreated: 1</code>	HTTP <code>201</code>
<code>orderCreated: 0</code> (already exists)	HTTP <code>200</code>
<code>productCreated</code> / <code>productUpdated</code> / <code>orderUpdated</code>	<i>removed</i>
<code>omsOrderItemIds: "8059,8060"</code> (comma-separated string)	<code>omsOrderItemIds: ["8059","8060"]</code> (array of strings)
<code>statusCode</code> / <code>message</code> in the body	HTTP status code; errors carry <code>error</code> + <code>code</code>
<code>data.status</code> , <code>data.trackingNumber</code> , <code>data.trackingDate</code>	<code>status</code> , <code>tracking[].trackingNumber</code> , <code>tracking[].dateShipped</code>
<code>data.serializedData</code> , <code>shippingProviderName</code>	<code>tracking[].provider</code>
—	<code>walletDebited</code> , <code>omsPushStatus</code> , <code>402</code> HOLD

Behaviour changes to review in your code

- 402 is not a failure.** The old API had no wallet layer. Handle `402` as "held, retry with the same reference" — never as "cancel the order".
- Retries are safe.** `externalRef` deduplicates; the old `orderCreated: 0` check is replaced by the `200` vs `201` distinction.
- Tracking moved to the order.** One `GET` per order replaces one `GET` per line item, and the result is an array.
- Product data is ours.** Stop sending titles and image URLs.