**Comestri V5 Ecommerce API documentation**

Professional Services

Date: 2021-06-10



Table of Contents

[1 Summary 3](#_Toc103006619)

[2 API integration guide 3](#_Toc103006620)

[2.1 Rate limits 3](#_Toc103006621)

[2.2 Authentication 3](#_Toc103006622)

[2.2.1 (Deprecated) Token + email 3](#_Toc103006623)

[2.2.2 Cognito 3](#_Toc103006624)

[2.2.3 Setting up API users in Comestri 3](#_Toc103006625)

[2.2.4 Calling Comestri APIs 4](#_Toc103006626)

[3 Interfaces for Ecommerce integration 5](#_Toc103006627)

[3.1 Products from Comestri 5](#_Toc103006628)

[3.1.1 Filters 6](#_Toc103006629)

[3.2 Orders into Comestri 6](#_Toc103006630)

[3.2.1 Order data structure 6](#_Toc103006631)

[4 Other interfaces 7](#_Toc103006632)

[4.1 Products into Comestri 7](#_Toc103006633)

[4.1.1 Bulk API vs Product API 7](#_Toc103006634)

[4.1.2 New products 7](#_Toc103006635)

[4.1.3 Updating products 8](#_Toc103006636)

# Summary

This document outlines the key APIs required to integrate an eCommerce platform with Comestri.

This document covers Comestri V5 CM, which is the current Comestri version as of 19 Jan 2021.

# API integration guide

## Rate limits

Comestri limits API calls to 3600 per hour.

When loading significant quantities of data into Comestri, it is recommended to use the Bulk Jobs APIs instead of individual ones, for two reasons. Firstly, the bulk job will appear in the Jobs section in Comestri, allowing Comestri users to monitor job progress and detect errors. Secondly, a bulk call counts as one API call, but can contain hundreds of items (such as products).

## Authentication

### (Deprecated) Token + email

*DEPRECATION WARNING: Starting end 2021 the current Username + Token authentication will be deprecated. External systems calling Comestri APIs will need to start using Cognito authentication.*

These can be retrieved by logging into Comestri, clicking on the tenant name on the top right, and clicking Authentication Token. Pass these values in the header of the API call.

### Cognito

All Comestri tenants can use api users with Cognito.

### Setting up API users in Comestri

API users can be configured in Comestri->Users->Manage API Accounts. The account must be created, and the upon saving the API credentials will appear once, and should be saved immediately. The API user must be given permission to access the API endpoint required.

### Calling Comestri APIs

In order to make API calls to Comestri, the Client must acquire a valid bearer token. API User bearer tokens expire after 60 minutes.

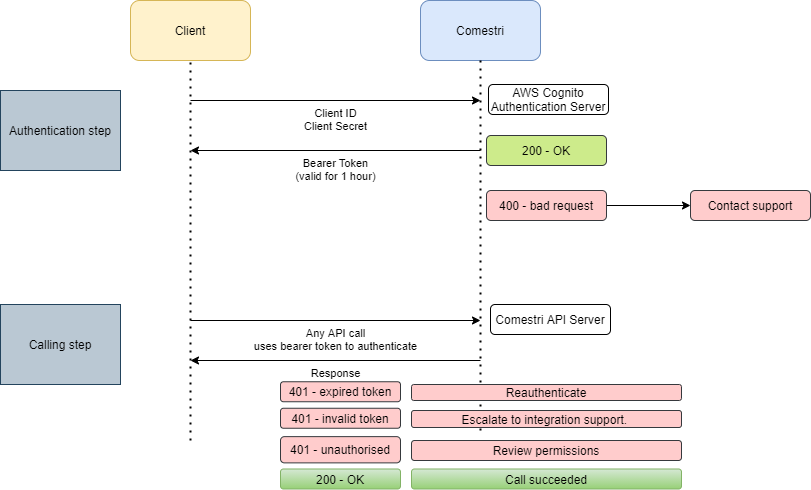
If you are using the Comestri postman collection you will not need to manually retrieve the token each time as it is built into the collections function.

Comestri APIs can be called by using a valid bearer token.

#### Cognito response codes and actions

|  |  |  |
| --- | --- | --- |
| API Response code | Meaning | Action |
| 200 - OK | Call succeeded | Continue |
| 401 - expired token | The bearer token has expired, a new token is required. | Reauthenticate - request a new token bearer token first, then try the API call again. |
| 401 - invalid token | API user isn’t configured to have access to the API call attempted. Permission must be enabled for this API call. | Fix API user permissions. Token access is managed in the API User Permissions section in Comestri Users.  If stuck, contact Comestri support to review API user permissions. |
| 401 - unauthorised | Mapping error - token format wrong, token not sent, etc. | Stop and escalate to integration support. |

#### API Process Flow



# Interfaces for Ecommerce integration

## Products from Comestri

Products and their related data are retrieved from Comestri.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data type | Notes | Type of export | API call | Parameters |
| General category structure | To retrieve the category hierarchy | Full | /api/v2/categories |  |
| General category structure | To retrieve the category hierarchy | Delta | /api/v2/categories | updated\_time\_from,updated\_time\_to |
| Product category | To retrieve the category assignments of individual products | Full | /api/v2/products | response\_associated\_fields=product\_categories |
| Product category | To retrieve the category assignments of individual products | Delta | /api/v2/products | response\_associated\_fields=product\_categories |
| Product images | To retrieve current images associated with a product. | Full | /api/v2/products | response\_associated\_fields=media\_items |
| Product images | To retrieve changes to images (including deletions) associated with a product | Delta | /api/v2/products | updated\_time\_from,response\_associated\_fields=media\_items |

### Filters

The Filter parameter can be added to some GET API calls.

e.g. entity\_id[\*].name['BARCODE'] IS NULL

Example call:

|  |
| --- |
| /api/v2/products?filter=entity\_id[\*].name['BARCODE'] IS NULL |

## Orders into Comestri

### Order data structure

The Comestri LOM (Legacy Order Manager) structures orders into a hierarchy. Each node in the hierarchy has a number of fields to populate, plus some sub nodes. The structure below contains all the sub nodes. Note that there are fields under most of these nodes which are note shown here:

* Header
* Source
* Shipping addresses
* Billing addresses
* Discounts
* Shipping Service
* Payments
* Additionals (not visible in Comestri)
* Taxes
* Buyer
* Personal Info (this node must be used when POSTING orders. This node isn’t visible when making the GET orders call.)
* Parties
* Additionals (not visible in Comestri)
* Order Lines
* Order line
* Discounts
* Additionals (not visible in Comestri)
* Discounts
* Taxes
* Options (not visible in Comestri, not always passed by each channel)

# Other interfaces

## Products into Comestri

### Bulk API vs Product API

It is recommended when pushing large updates to always use the Bulk Jobs API to pass products into Comestri.

### New products

An entire product hierarchy can be passed into Comestri in a single product node. For instance, where a 3 class Style->Colour->Size hierarchy exists, the node can be constructed by adding size products as variants to their colour products, which are added as variants to the style product.

See example here: <https://product.comestri.com/apidoc/2.0/products/create.html> (scroll to the heading: Add three level products (Style-Colour-Size) or more along with associations)

### Updating products

#### Recommended approach

It is recommended to pass an update to the specific object being changed only.

When adding a new product to an existing structure (e.g. a new size to a colour or a new colour to a style) create the product and link it to the parent class.

#### Efficient design and product deletions

While product updates can be passed using the whole product hierarchy, this design pattern is not recommended as it results in a large amount of superfluous data processing.

It is recommended not to delete products in Comestri before removing them from channels. Often clients choose to inactivate a product and then delete it some weeks or months later.