

# Advanced Metafields App User Guide

## TABLE OF CONTENTS

<b>Getting Started</b>	<b>2</b>
<b>Metafields Schemas</b>	<b>2</b>
Metafield Schema Types and Available Configurations	3
Text	3
Text Area	4
Number	4
Boolean	4
Select	5
Email	5
Phone	5
URL	5
Date	6
Color	6
WYSIWYG	6
HTML	6
JSON	6
Video	7
<b>Metafields</b>	<b>7</b>
Fetching in Stencil Storefront using GraphQL	9
Fetching using GraphQL Storefront API	11
Fetching using BigCommerce API	11

## Getting Started

To get started, you must first install the **Advanced Metafields App by Ecommerce Store Apps**.

1. Go to the [Bigcommerce Marketplace](#) and search for Advanced Metafields by Ecommerce Store Apps.
2. Click “Get this App”.
3. Log in to your store and then click the “Install” button.
4. You’ll be redirected to the page with a list of permissions needed for the app to function properly. We only ask for the permissions we need. Click on the “install” button.
5. When the app successfully installs, you’ll be redirected to the dashboard.

You’ll need to know about Schemas and Metfields to get the most out of the app. Let’s learn how to get it up and running!

## Metafields Schemas

Metafields Schemas are what determine what type of input fields will appear in each catalog list type metafield’s page. Schemas are linked to catalog types, namely: Products, Variants, Categories, Brands, Orders, and Channels. Schemas are not shared among catalog types. For example, a schema that exists in Product will not appear in a Variant’s metafield page. You can create Metafield Schemas by clicking on one of the catalog types in the dashboard.

---

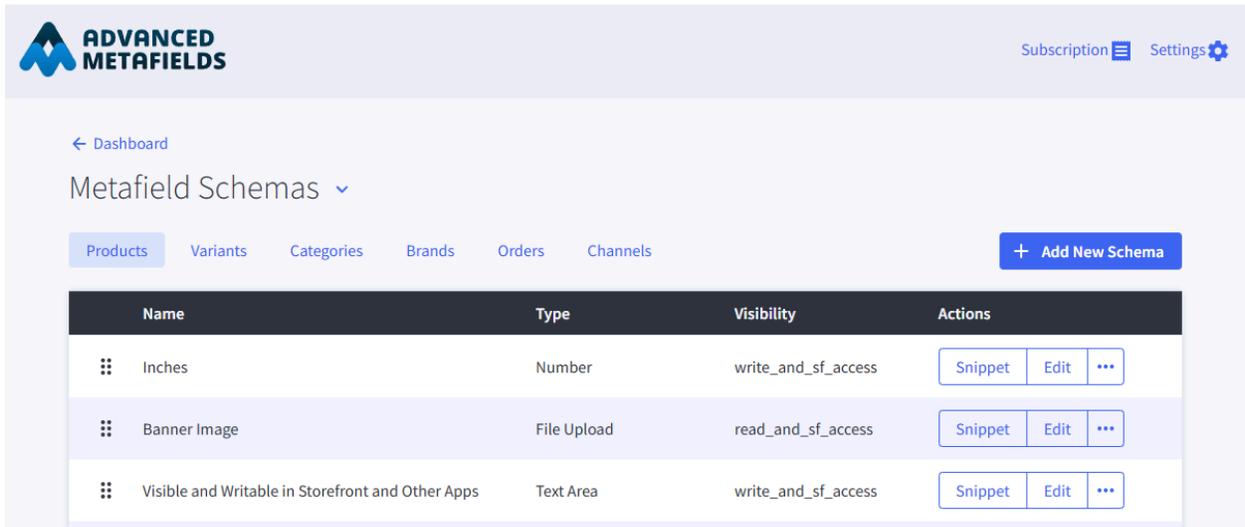
### Metafield Schemas

Schemas are the inputs that will appear in the Catalog’s Metafields page to create and update the metafields.

Products >	Variants >
Categories >	Brands >
Orders >	Channels >

---

In this page all existing schemas will be listed out, you can create new schemas, and edit or delete existing schemas. To create a new schema, just click on the “Add new Schema” button.



The screenshot shows the 'Advanced Metafields' dashboard. At the top, there's a navigation bar with 'Subscription' and 'Settings'. Below that, a breadcrumb trail shows 'Dashboard' and 'Metafield Schemas'. A filter bar includes 'Products', 'Variants', 'Categories', 'Brands', 'Orders', and 'Channels', along with an 'Add New Schema' button. The main content is a table with the following data:

Name	Type	Visibility	Actions
Inches	Number	write_and_sf_access	Snippet Edit ...
Banner Image	File Upload	read_and_sf_access	Snippet Edit ...
Visible and Writable in Storefront and Other Apps	Text Area	write_and_sf_access	Snippet Edit ...

When clicked, a popup will appear where you'll have to input the following information:

- **Name:** This is only for display in the app.
- **Description:** This is a brief summary of what the metafields represent.
- **Namespace:** This is used for querying metafields. Every namespace/key pair must be unique. This app's default namespace is "ct\_metafields."
- **Key:** This is the key from which you'll access this metafield in the theme or using the API.
- **Visibility:** This determines if this metafield is able to be fetched from outside the app or if it will be used internally only.
- **Type:** This is the type of input that will appear in the metafield page.

When all inputs are selected, click the "Save" button. This catalog item will now have your Metafield Schema, and it is ready to be used in the metafields page.

Bellow you can find a list of types that are available and their available configurations.

## Metafield Schema Types and Available Configurations

### Text

A normal HTML text input.

#### Available configurations:

**Minimum Length:** The minimum length of characters the text input is required to have when creating the metafield.

**Maximum Length:** The maximum length of characters the text input is required to have when creating the metafield. The maximum length possible is 65535 characters, limited by the native size of the metafield.

### Text Area

A normal HTML text input.

#### Available configurations:

**Minimum Length:** The minimum length of characters the text input is required to have when creating the metafield.

**Maximum Length:** The maximum length of characters the text input is required to have when creating the metafield. The maximum length possible is 65535 characters, limited by the native size of the metafield.

### Number

A normal HTML number input.

#### Available configurations:

**Accepted Values:** What kind of number is accepted in the input, either a whole number or a decimal number.

**Minimum Value:** The minimum number value of this field. By limit of the metafield, the minimum value is negative 65535 long number of 9s or, more precisely,  $-(10^{65535}-1)$ .

**Maximum Value:** The maximum number value of this field. By limit of the metafield, the max value is a 65535 long number of 9s or, more precisely,  $10^{65535}-1$ .

**Prefix:** A text that will appear before the input. It will not affect the value, which will only be the number. Example: US\$, CAD\$, etc.

**Suffix:** A text that will appear after the input. It will not affect the value, which will only be the number. Example: Kg, Lb, etc.

### Boolean

A switch with 2 possible values, either true or false, or custom values.

#### Available configurations:

**Default Value:** If it is either true or false by default.

**False Label:** What is written on the left side of the switch.

**True Label:** What is written on the right side of the switch.

**False Value:** The value of false, if empty it is simply false.

**True Value:** The value of true, if empty it is simply true.

## Select

An input with various custom options and display types, where you can choose one of the options.

### Available configurations:

**Display Type:** How the input will be displayed. Either a select dropdown, a list of radio buttons, or a list of checkboxes.

**Default Option:** The default value to be chosen based on one of the user created options.

**Options:** A user-created list of options with a label and value each.

## Email

An HTML email input.

### Available configurations:

None.

## Phone

An input for phone numbers.

### Available configurations:

None.

## URL

A HTML text input that only accepts urls as valid input.

**Available configurations:**

**Https Only:** If the input accepts only https urls.

**Date**

A datetime input, with a calendar and optional time input.

**Available configurations:**

**Date only or Date and Time:** If there will only be a date calendar input or date and time input.

**Time Format:** If it should be a 12-hour or 24-hour format.

**Color**

An input with a color picker for easily choosing colors.

**Available configurations:**

None.

**WYSIWYG**

A WYSIWYG editor allowing the user to input structured content.

**Available configurations:**

None.

**HTML**

A textarea input with html highlighting.

**Available configurations:**

None.

**JSON**

A textarea input with json validation.

**Available configurations:**

None.

## Video

A text input to receive a video link.

### Available configurations:

None.

## Metafields

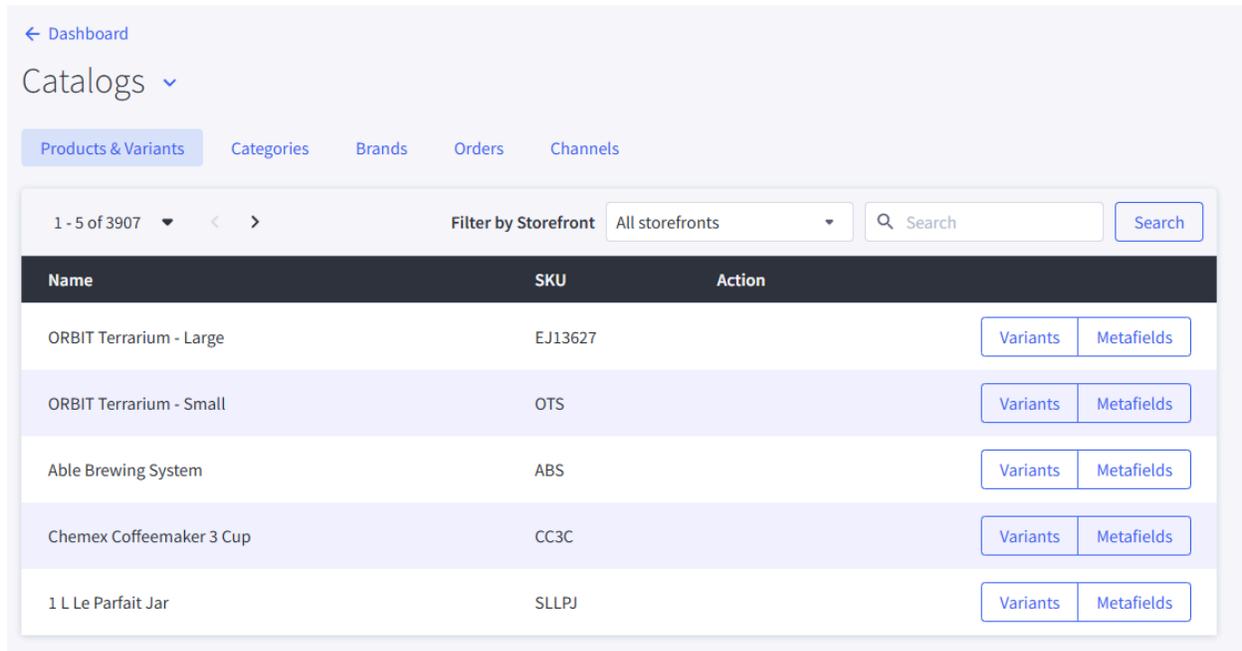
To create the actual metafields, you first need to access a catalog list. You can do this by clicking on one of the links under “Catalog Lists” in the dashboard.

### Catalog Lists

Access the metafields using the catalog lists below. There you will be able to create and update the metafields on these items using the schemas created previously.

<a href="#">Products &amp; Variants</a> >	<a href="#">Categories</a> >
<a href="#">Brands</a> >	<a href="#">Orders</a> >
<a href="#">Channels</a> >	

Here you'll be presented with a list of items of that specific type. Next to each item, you'll find a "Metafields" button that, when clicked, will take you to the metafields page.



← Dashboard

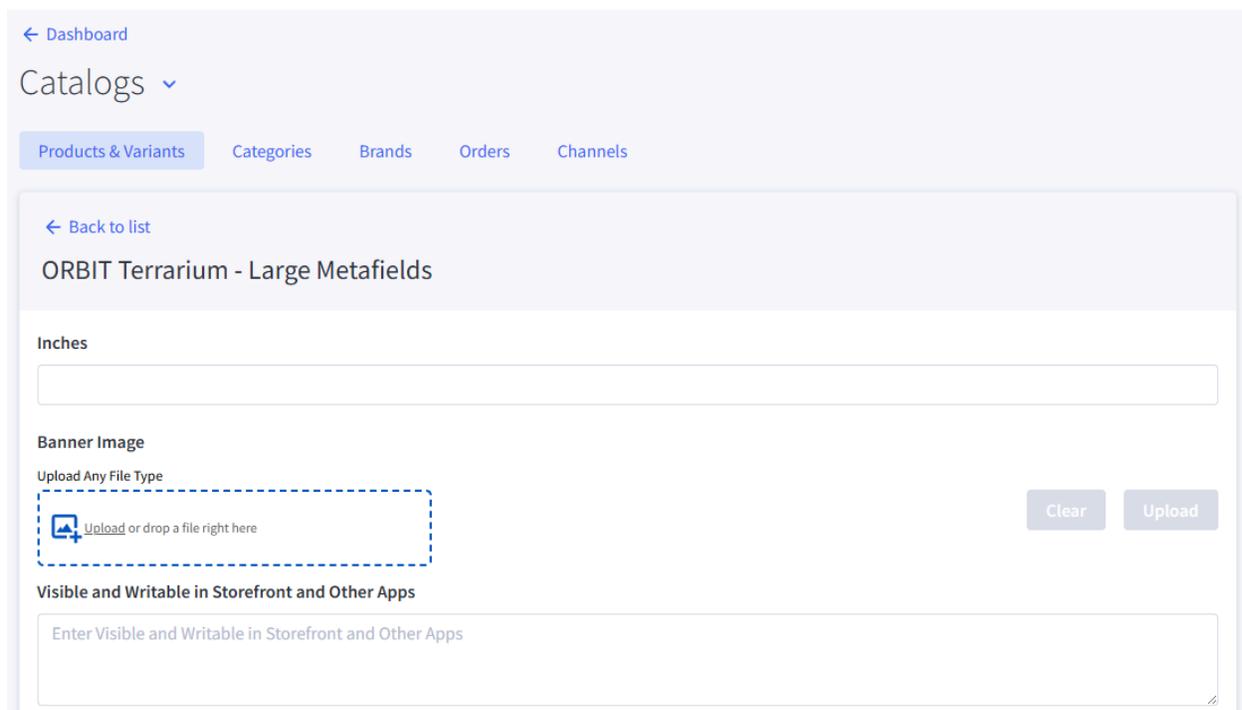
Catalogs ▾

Products & Variants Categories Brands Orders Channels

1 - 5 of 3907 < > Filter by Storefront All storefronts Search Search

Name	SKU	Action
ORBIT Terrarium - Large	EJ13627	<a href="#">Variants</a> <a href="#">Metafields</a>
ORBIT Terrarium - Small	OTS	<a href="#">Variants</a> <a href="#">Metafields</a>
Able Brewing System	ABS	<a href="#">Variants</a> <a href="#">Metafields</a>
Chemex Coffeemaker 3 Cup	CC3C	<a href="#">Variants</a> <a href="#">Metafields</a>
1 L Le Parfait Jar	SLLPJ	<a href="#">Variants</a> <a href="#">Metafields</a>

Here you'll see all your previously created schemas and be able to create, edit, and delete metafields.



← Dashboard

Catalogs ▾

Products & Variants Categories Brands Orders Channels

← Back to list

ORBIT Terrarium - Large Metafields

Inches

Banner Image

Upload Any File Type

Visible and Writable in Storefront and Other Apps

From here, the metafield can be fetched using the API and GraphQL and used as needed.

## Fetching in Stencil Storefront using GraphQL

To fetch metafields using GraphQL in the stencil storefront, we can use a GraphQL query in the frontmatter section of the page in the code. Let's look at an example using the product page.

The product page in the cornerstone theme is found at `templates\pages\product.html`. Here, at the top of the code we have the frontmatter section:

```
---
product:
  videos:
    limit: {{theme_settings.productpage_videos_count}}
  reviews:
    limit: {{theme_settings.productpage_reviews_count}}
  related_products:
    limit: {{theme_settings.productpage_related_products_count}}
  similar_by_views:
    limit: {{theme_settings.productpage_similar_by_views_count}}
---
```

Here you can add the graphql query using the “gql” attribute:

```

---
product:
  videos:
    limit: {{theme_settings.productpage_videos_count}}
  reviews:
    limit: {{theme_settings.productpage_reviews_count}}
  related_products:
    limit: {{theme_settings.productpage_related_products_count}}
  similar_by_views:
    limit: {{theme_settings.productpage_similar_by_views_count}}
gql: "query getProductMetafields($productId: Int!) {
  site {
    product(entityId: $productId) {
      metafields(namespace: 'ct_metafields', keys: ['author']) {
        edges {
          node {
            key
            value
          }
        }
      }
    }
  }
}"
---

```

The metafields created in the app will always be in the namespace “ct\_metafields”. Here you can fetch as many metafields as you want, listing their keys in the “keys” array. Wherever you want to use the value of a specific metafield, you can fetch it using the following snippet:

```

{{#each gql.data.site.product.metafields.edges}}
  {{#with node}}
    {{#if name '==' 'author'}}
      {{value}}
    {{/if}}
  {{/with}}
{{/each}}

```

## Fetching using GraphQL Storefront API

To learn more about fetching using GraphQL Storefront API, but not in front matter, please refer to the following:

<https://developer.bigcommerce.com/docs/graphql-storefront#graphql-storefront-api>

## Fetching using BigCommerce API

To fetch using the BigCommerce API, you can make a request to the related item's endpoint.

Refer to the following links:

- [Products](#)
- [Product Variants](#)
- [Categories](#)
- [Brands](#)
- [Orders](#)
- [Channels](#)