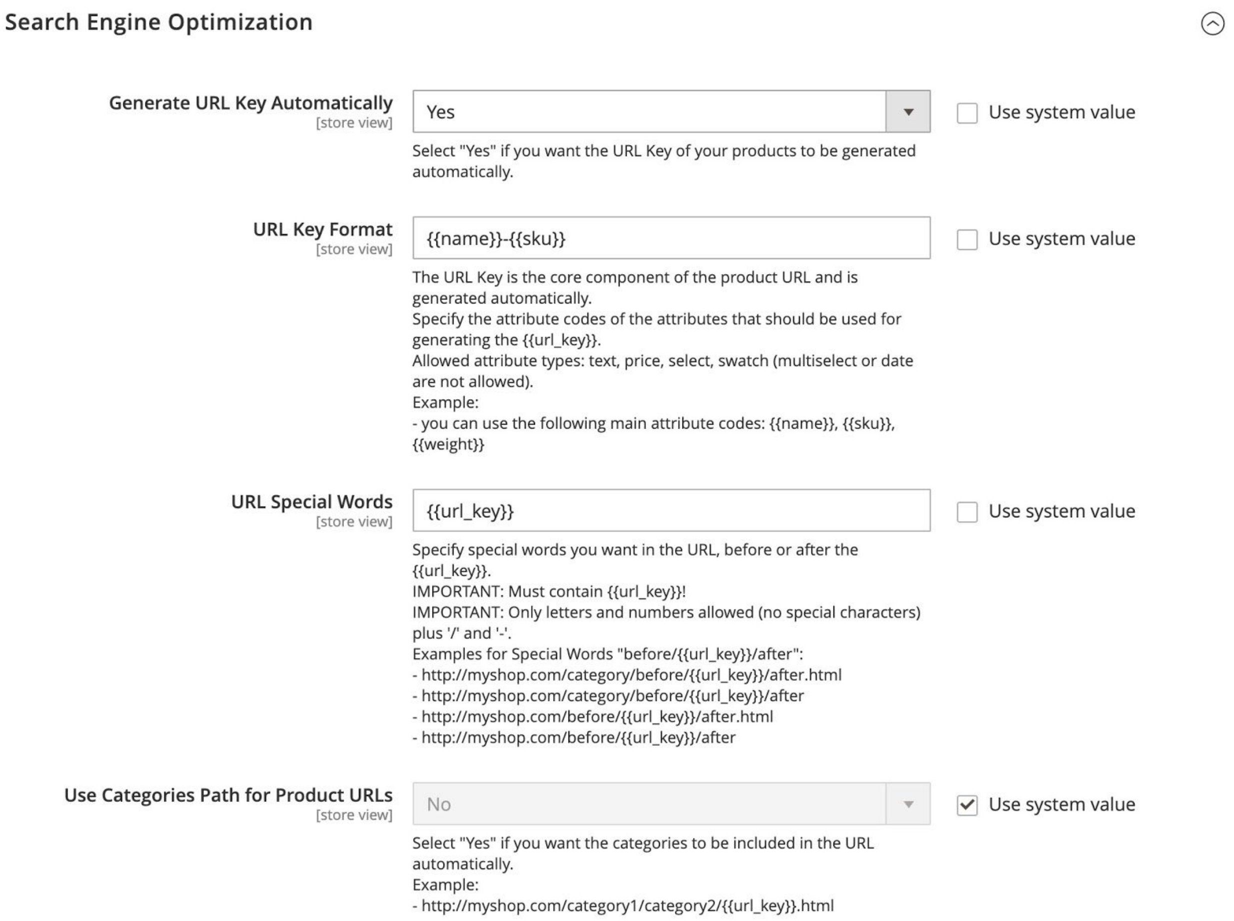
WebPanda

URL Rewrites Regenerate and Customize

Brief Content:

I. General Settings

To configure the extension go to **Stores** → **Configuration** → **Catalog** → **Catalog** → **Search Engine Optimization**



Search Engine Optimization

Generate URL Key Automatically [store view]: Yes □ Use system value

Select "Yes" if you want the URL Key of your products to be generated automatically.

URL Key Format [store view]: {{name}}-{{sku}} □ Use system value

The URL Key is the core component of the product URL and is generated automatically.
Specify the attribute codes of the attributes that should be used for generating the {{url_key}}.
Allowed attribute types: text, price, select, swatch (multiselect or date are not allowed).
Example:
- you can use the following main attribute codes: {{name}}, {{sku}}, {{weight}}

URL Special Words [store view]: {{url_key}} □ Use system value

Specify special words you want in the URL, before or after the {{url_key}}.
IMPORTANT: Must contain {{url_key}}!
IMPORTANT: Only letters and numbers allowed (no special characters) plus '/' and '-'.
Examples for Special Words "before/{{url_key}}/after":
- http://myshop.com/category/before/{{url_key}}/after.html
- http://myshop.com/category/before/{{url_key}}/after
- http://myshop.com/before/{{url_key}}/after.html
- http://myshop.com/before/{{url_key}}/after

Use Categories Path for Product URLs [store view]: No ☑ Use system value

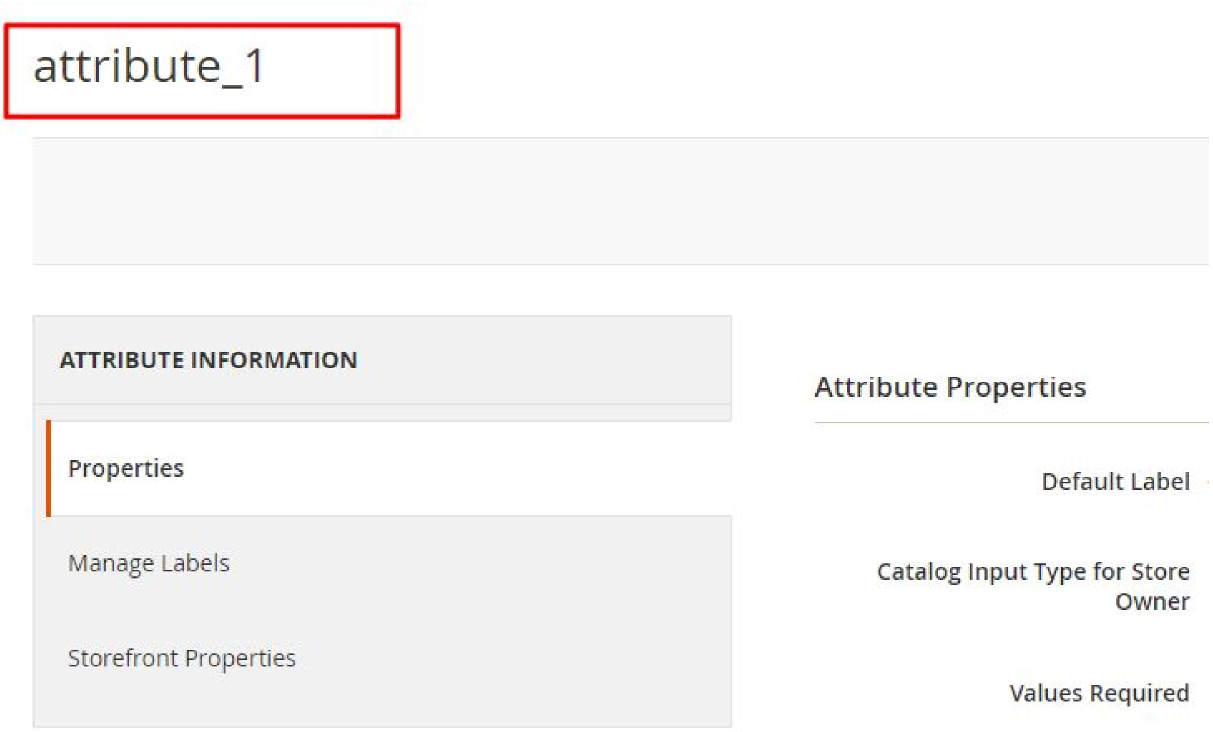
Select "Yes" if you want the categories to be included in the URL automatically.
Example:
- http://myshop.com/category1/category2/{{url_key}}.html

Generate URL Key Automatically:

- Select "Yes" if you want the URL Key of your products to be generated automatically.
- Select "No" and you will be able to set the URL Key the way you need it and regenerate the URL rewrites based on this URL Key

URL Key Format:

- The **URL Key** is the core component of the product URL. This field allows you to choose which attributes you want to use for generating the format of all product URLs.
- For every attribute you want to add to the URL, you need to specify its **Attribute Code** between **{{** and **}}**. Every time the extension will regenerate the URL for a product it will look for the product value for each attribute used in this field and add it to the URL.
- The separators you are allowed to use are '/' and '-'.
- Allowed attribute type are: text, price, select, swatch.
- After changing this value you need to run the URL Regeneration for the change to take effect.
- The default value for this field is "{{name}}-{{sku}}".
- To find the attribute code of an attribute go to **Stores** → **Product** then select the attribute you want and copy the value from the top, like in the following screenshot:



Use Categories Path for Product URLs:

- Select "Yes" if you want the product categories names to be included in the product URL.
- After changing this value you don't need to run the URL Regeneration. You only need to refresh the cache.
- The default value for this field is "No".

URL Special Words:

- This field allows you to add custom keywords in the product URL before or after the {{url_key}}. You can add for example "product/{{url_key}}".
- The **URL Special Words** field needs to always contain the "{{url_key}}".
- The separators you are allowed to use are '/' and '-'.
- After changing this value you need to run the URL Regeneration for the change to take effect.
- The default value for this field is "{{url_key}}".

Product URL Suffix:

- Choose if you want to allow a suffix for the product URL.
- After changing this value you need to run the URL Regeneration for the change to take effect.
- The default value for this field is ".html".

II. How the URL generation for one product works:

For a product with Name **"Blue Shoe"**, SKU **"123456"** and assigned to category **"shoes-category"**, the algorithm will follow these steps:

1. The **URL Key Format** field is parsed. The value we get represents the **{{url_key}}** that will be used in the **URL Special Words** field.
   Let's say for example we have the value **{{name}}-{{sku}}**.
   This leads to the **{{url_key}}** final value **"blue-shoe-123456"**
2. Next the **URL Special Words** field is parsed.
   Let's say for example we have the value **"product/{{url_key}}"**.
   This leads to the URL **"product/blue-shoe-123456"**
3. Next the **Product URL Suffix** field is parsed.
   Let's say for example we have the value **".html"**.
   This leads to the URL **"product/blue-shoe-123456.html"**
4. Next the **Use Categories Path for Product URLs** field is parsed.
   - If you have selected **"Yes"** then the final URL of the product will be:
     shoes-category/product/blue-shoe-123456.html
   - If you have selected **"No"** the final URL of the product will be:
     product/blue-shoe-123456.html

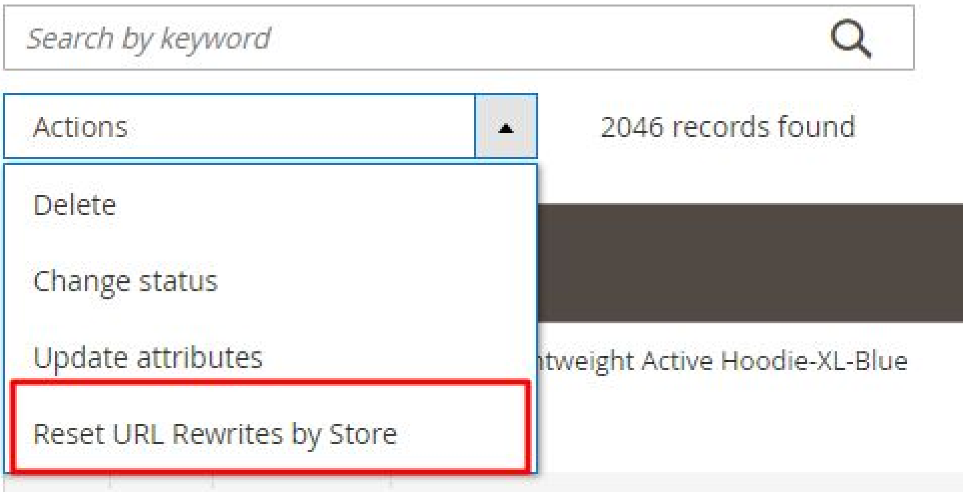
III. Product URL Regeneration

After doing all the configuration you can manage the product URL generation and regeneration.

When creating/updating products from Admin Form, API Import and by CSV Import they will have the URL generated automatically based on the configuration above. Regenerating the URLs for them won't be necessary.

If you only need to reset the URLs for **existing products** you can use the following 2 options:

1. Products Grid Mass Action



- Go to **Catalog** → **Products**
- Select the list of products for which you want to reset the URLs. You also have the option to select **All Products on This Page** or **All Products** on the website.
- **IMPORTANT:** Selecting a large number of products (over 1000) could lead to your server responding with timeout. The reason for this is that the PHP is configured by default to respond with this kind of error if running a script takes more than 30 seconds. You should select maximum 1000 products at a time when using this method of regenerating the product URLs, or change the PHP settings on your server to extend the “max_execution_time” to more then the default 30 seconds.
- After selecting the products you want to reset, select “Reset URL Rewrites by Store”. You will be asked to select the stores for which you want to reset the product URLs.
- **IMPORTANT:** If the product value for at least one attribute (used in the **URL Key Format**) is different for several stores, you will need to run the URL Regeneration for each store.
- After all product URLs are regenerated you need to **Clear Cache**. Go to **System** → **Cache Management.**

2. CLI Command

This is the prefered method for resetting large numbers of products at once. This is because most PHP Servers are configured to allow a script to run 30 seconds at most, and return a timeout error.

How to use:

```
php bin/magento webpanda:product-url:regenerate [--store="..."] [--startPid="..."] [--endPid="..."] [pids1] ... [pidsN]
```

It allows you to select:

- the store for which to reset the URLs. Leave **empty** and it will use **Default Store**
- an interval of products IDs to process
- a list of product IDs for which to reset the URLs. Leave **empty** to reset all products

Examples:

1. php bin/magento webpanda:product-url:regenerate --store 1 1001 1002 1003 1004
   - this will reset those **4 products** only for **store 1**
2. php bin/magento webpanda:product-url:regenerate --store 2
   - this will reset **all products** in **store 2**
3. php bin/magento webpanda:product-url:regenerate --store 2 --startPid 1 -- endPid 100
   - this will reset products with IDs starting from **1 to 100** for **store 2**
4. php bin/magento webpanda:product-url:regenerate
   - this will reset **all products** on the website
   - IMPORTANT: If the product value for at least one attribute (used in the **URL Key Format**) is different for several stores, you will need to run the URL Regeneration for each store.

After the command finishes resetting all products you will need to clear the cache either from **System → Cache Management** or by command line: **php bin/magento cache:flush**

Important: **If you have another module that rewrites functionalities in the Product CSV Import, the URL generation functionalities might not work on the Product Import. In this case you will need to regenerate the product URLs after you import the products from CSV.**

3. CLI Commands for regenerating Category URLs

These commands will regenerate category URL based on its url_path and url_path based on its url_key and its parent categories.
The first command is used to make sure that the URL problems of your categories is not caused by some category attribute update malfunction or maybe by a migration from magento 1 to magento 2(this happens often)

How to use:

php bin/magento webpanda:category-path:regenerate [--store="..."] [cids1] ... [cidsN]

php bin/magento webpanda:category-url:regenerate [--store="..."] [cids1] ... [cidsN]

Important: **If your website has multiple stores, you need to run both commands for each individual store ID (except for store ID 1 because this is the default one)**

4. CLI Commands for regenerating Product Metadata

Using this command you can regenerate the metadata fields automatically with the data from the product name and product short description.

How to use:

```
php bin/magento webpanda:product-metadata:regenerate [--store="..."] [--startPid="..."] [--endPid="..."] [pids1] ... [pidsN]
```