

Smile

OPEN SOURCE SOLUTIONS

ElasticSuite *for Magento 2*

User Guide

Versions: ElasticSuite 2.0 to 2.1

Date: 6th July 2016

Author: Pierrick Olivier

open
source

TABLE OF CONTENT

I - INTRODUCTION	3
I.1 WHAT IS ELASTICSUITE?	3
I.2 WHO WE ARE	3
I.3 REQUIREMENTS	3
I.4 TERMINOLOGY	4
I.5 FOR MORE INFORMATION	4
II - SEARCH ENGINE	5
II.1 FUZZY SEARCH	5
II.2 PHONETIC SEARCH	6
II.3 CUSTOMIZABLE THESAURUS	8
II.3.a Thesaurus settings	9
II.3.b Synonyms	10
II.3.c Expansions	11
II.4 AUTOMATIC STOPWORDS DETECTION	13
II.5 PHRASE MATCHING	13
II.6 MINIMUM NUMBER OF WORDS MATCHING THE QUERY	15
II.7 TIE BREAKER: COMBINATION OF SEVERAL PRODUCT ATTRIBUTES	16
II.8 BETTER AUTOCOMPLETE	16
III - FILTERS	18
III.1 MULTIPLE SELECT IN LAYERED NAVIGATION	18
III.2 ABILITY TO SORT VALUES IN FILTERS	18
III.3 ENHANCED PRICE SLIDER	19
III.4 SUPPORT OF LARGE CATALOGS	19
IV - MERCHANDISING	21
IV.1 VIRTUAL CATEGORIES	21
IV.2 SORTING PRODUCTS BY DRAG & DROP	23
V - TECHNICAL IMPROVEMENTS	25

I - INTRODUCTION

I.1 WHAT IS ELASTICSUITE?

ElasticSuite is a set of modules for Magento that provides several enhancements to Magento:

- | A powerful and customizable search engine
- | Improvements on the layered navigation
- | New merchandising features
- | High performances & scalability

This set of modules has been designed to help merchants take up one of the biggest challenges in e-commerce: how to display the good product at the right time to the customers?

This set of modules has been created by Smile Lab and is released under the Open Software Licence version 3.0.

I.2 WHO WE ARE

Set up in 1991, Smile is the first European integrator specialized in open source solutions.

On top of our engineering expertise, Smile offers a vast range of services: consulting upstream and in support of the projects, the interactive agency both in creation and web design and in editorial, strategic and e-marketing advice, application maintenance, training, support & maintenance in operational condition, hosting and exploitation.

Magento recognized Smile as a core partner through the «Gold» partnership badge.



Since 2014, Smilelab is the innovation and experimentation entity of Smile. Our multidisciplinary team brings together experts in technology, innovation, and new applications. Together we explore, invent, and test technologies of the future, to better serve our clients.

I.3 REQUIREMENTS

ElasticSuite has been successfully deployed and tested on the following Magento versions:

- | Magento EE 2.0
- | Magento CE 2.0

Another version has been released for Magento 1.

As far as possible, you should install the module at the early beginning of your Magento project, at the same time you install Magento. This way, your development team won't spend additional time to handle ElasticSuite specificities.

1.4 TERMINOLOGY

Product attribute	A product attribute is a field in the product form. Product attributes can be managed in the back-office in the menu Stores > Attributes > Product.
Filter	On most e-commerce website, when a list of products is displayed, several filters are displayed in the left columns to help the user to find a product: filter by price, brand, category...
Facet	Synonym for « Filter ».
Layered Navigation	The « layered navigation » is the use of filters for navigation.
Stopword	A stopword is a word very frequently used and with little meaning for a search. For example: and, or, to, for, are, be, by, in,...
Boost	The ranking of a search engine is based on the calculation of a relevance score for each product. Applying a boost of +10% to some products means to increase the score of these products by +10%.
Fuzzy search	A fuzzy search allows small mistakes during typing, by allowing small changes in the letters entered by the user.
Phonetic search	The phonetic search allows spelling and grammatical errors during typing, by allowing changing a word by another with a similar sound.
Autocomplete	We call « Autocomplete » the suggestions displayed during typing in the search field.

1.5 FOR MORE INFORMATION

ElasticSuite official website	http://magento-elastic-suite.io
GitHub ElasticSuite for Magento 2	https://github.com/Smile-SA/elasticsuite
GitHub ElasticSuite for Magento 1	https://github.com/Smile-SA/smile-magento-elasticsearch
Demo store (Magento 2)	http://demo.magento-elastic-suite.io
Smile official website	http://www.smile.fr

II - SEARCH ENGINE

II.1 FUZZY SEARCH

Feature added
in version 2.0.0

Fuzzy search is a way to correct mistakes during typing, using the « Levenstein distance ». With the Levenstein distance, the similarity of two words are evaluated by the number of letters that you should remove, add, or replace in order to change one word into another. The more the words are similar, the more ElasticSuite will boost the product.




Example

If a user enters the query « **trouper** », then we can assume that there is a high probability that the user is looking for a « **trouser** », because only 1 letter should be replaced in order to change the word « trouper » into « trouser ».

In comparison, there is a lower probability that the user is looking for « **trainers** » because 4 letters needs to be replaced/added in order to change « trouper » into « trainers ».



If all words in the request of the user return an exact match, then an exact search is performed without fuzzy search. Otherwise, if at least one word doesn't provide exact results, then a fuzzy search is performed for the entire query and an information banner is displayed to the user:

 No search results for: 'trouper'. We propose you approaching results.

The fuzzy search can be enabled in the back-office:

1. Open the menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Spellchecking configuration > Search Fuzziness configuration**

Search fuzziness configuration

Enable fuzziness	<input type="text" value="Yes"/>	[CONTAINER - STORE VIEW]
Fuzziness value	<input type="text" value="AUTO"/> <small>Allowed values "0", "1" or "AUTO". See doc here for more information.</small>	[CONTAINER - STORE VIEW]
Fuzziness Prefix Length	<input type="text" value="1"/> <small>An integer between greater or equal than 0. See doc here for more information.</small>	[CONTAINER - STORE VIEW]
Max. expansion	<input type="text" value="10"/> <small>An integer between greater or equal than 0. See doc here for more information.</small>	[CONTAINER - STORE VIEW]

3. Change the following settings:

- a. **Enable fuzziness:** Enable/disable fuzzy search
- b. **Fuzziness value:** Maximum number of letters that may be replaced to change one word into another. The allowed values are: 1, 2, and AUTO (note: the help message displayed on the screenshot above is not correct).

If this setting is set to « AUTO » then the search engine allows 1 wrong letter for words with 3 to 5 letters, and 2 wrong letters for words with 6 letters or more.
- c. **Fuzziness prefix length:** Number of letters at the beginning of the word that should not be changed (fuzzy search is not applied to the prefix). A lower value will provide better search results but will increase the workload on the server.
- d. **Fuzziness max. expansions:** Maximum depth of the fuzzy search algorithm. A higher value will search for a higher number of differences of letters but will increase the workload on the server.

Be careful: the query could be very heavy if **Fuzziness prefix length** is set to **0** and **Fuzziness max expansions** is set to a high number.



Recommended values

Examples of values that provided good results on past projects:

Fuzziness value: AUTO
Fuzziness prefix length: 1
Fuzziness max. expansions: 10

Then you have to enable the spellcheck property on each product attribute used by the search engine:

- 1. Open the menu **Stores > Attributes > Products**
- 2. Select an attribute used by the search engine (the name, for example)
- 3. Open the tab **Storefront Properties**
- 4. Set **Used in spellcheck** to **Yes**

Used in spellcheck



Recommended values

It is recommended to enable spellcheck for all text attributes used by the search engine.

II.2 PHONETIC SEARCH

*Feature added
in version 2.0.0*

Phonetic search is a way to correct spelling and grammatical errors.

When phonetic search is enabled, the search engine will suggest products with a similar phonetic sound. The phonetic search can include fuzzy search in order to allow minor differences in the pronunciation.

In comparison with fuzzy search, the phonetic search allows more wrong letters, but fewer differences in the pronunciation.




Example

With fuzzy search, the query « **llaguing** » will produce no result or bad results.

With phonetic search, the query « **llaguing** » will return « **leggings** » products.

The Phonetic search is based on an algorithm called « Beider-Morse ». The list of supported languages is: English, French, German, Spanish, Italian, Russian, Hungarian, Romanian, and Turkish. The language used for phonetic search is the language of the Magento store view.

If all words in the request of the user return an exact match, then an exact search is performed without phonetic search. Otherwise, if at least one word doesn't provide exact results, then a phonetic search is performed for the entire query and an information banner is displayed to the user:

 No search results for: '**llaguing**'. We propose you approaching results.

The phonetic search can be set up in the back-office:

1. Open the menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Spellchecking configuration > Phonetic search configuration**

Enable phonetic search	<input type="text" value="No"/>	[CONTAINER - STORE VIEW]
	Require Phonetic analysis plugin installation.	
Enable phonetic fuzziness	<input type="text" value="Yes"/>	[CONTAINER - STORE VIEW]
Phonetic fuzziness value	<input type="text" value="AUTO"/>	[CONTAINER - STORE VIEW]
	Allowed values "0", "1" or "AUTO". See doc here for more information.	
Phonetic fuzziness prefix length	<input type="text" value="1"/>	[CONTAINER - STORE VIEW]
	An integer between greater or equal than 0. See doc here for more information.	
Phonetic fuzziness max expansion	<input type="text" value="2"/>	[CONTAINER - STORE VIEW]
	An integer between greater or equal than 0. See doc here for more information.	

3. Change the following settings:
 - a. **Enable phonetic search:** enable/disable phonetic search.
 - b. **Enable phonetic fuzziness:** enable/disable the fuzzy search in phonetic search.
 - c. **Phonetic Fuzziness value:** maximum number of phonemes that can be replaced when changing a word into another. The allowed values are: 1, 2, and AUTO (note: the help message displayed on the screenshot above is not correct).

If this setting is set to « AUTO » then the search engine will adjust the number of wrong phonemes allowed with the length of the word.

- d. **Phonetic Fuzziness prefix length:** Number of phonemes at the beginning of the word that should not be changed (no fuzzy search in the prefix). A lower value will provide better search results but will increase the workload on the server.
- e. **Fuzziness max expansions:** the maximum number of iterations in the fuzzy search algorithm. A higher value will search for a higher differences of sounds but will increase the workload on the server.

Be careful, the query could be very heavy if **Phonetic fuzzyness prefix length** is set to 0 and **Phonetic fuzziness max expansions** is set to a high number.



Recommended values

Examples of values that provided good results on past projects:

Enable phonetic search: Yes
Enable phonetic fuzziness: No

Phonetic fuzziness should be used carefully because it may return results a bit far from the request of the user.

Then you have to enable the spellcheck property on each product attribute used by the search engine:

1. Open the menu **Stores > Attributes > Products**
2. Select an attribute used by the search engine (the name, for example)
3. Open the tab **Storefront Properties**
4. Set **Used in spellcheck** to **Yes**

Used in spellcheck



Recommended values

It is recommended to enable spellcheck for all text attributes used by the search engine.

II.3 CUSTOMIZABLE THESAURUS

*Feature added
in version 2.1.0*

Users do not always formulate search queries using the best terms. Not “the best” means that the database does not contain the user entered terms and that the query may returns no result.

To prevent such issue, ElasticSuite search engine invoke a customizable thesaurus to increase the quality of the search results. The customizable thesaurus is considered as a meta-level process that is used to add more information to clarify the user's query. It is the process of rebuilding new informed queries from an existing one in order to improve the retrieval performance and help in matching additional documents.

The thesaurus can be used for:

- 1 Finding synonyms of words, and searching for the synonyms as well



Example

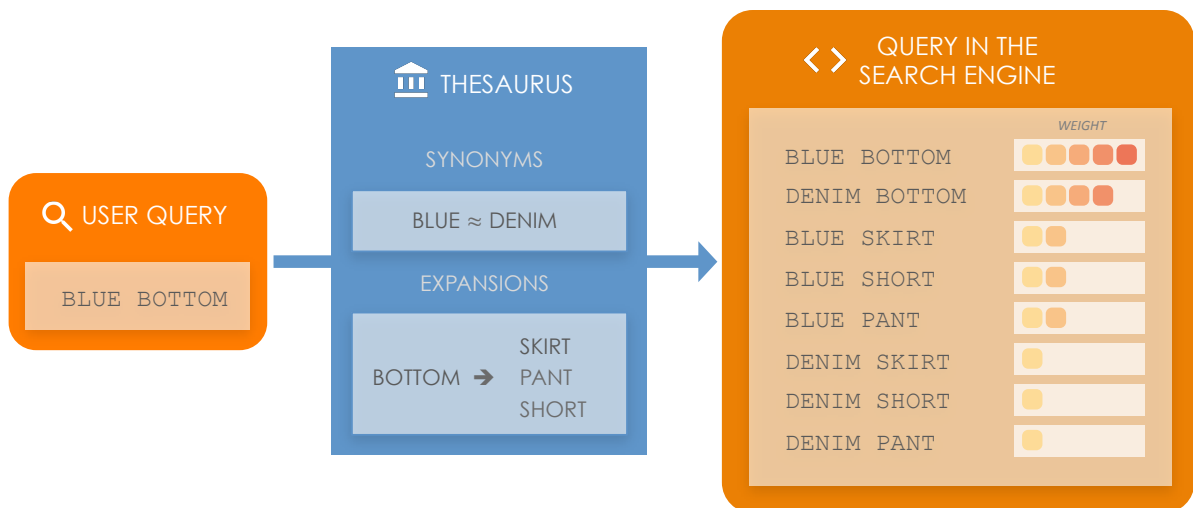
The word « denim » could be set as a synonym of « blue ».

- 2 Finding all the various morphological forms of words by expanding each word in the search query



Example

The word « bottom » could be expanded into « skirt », « pant », « short ».



Note

On a technical point of view, the thesaurus doesn't generate multiple requests to the database. For each user query, all synonyms and expansions are sent to the database into one single request, ensuring the best performances.

11.3.a Thesaurus settings

First you have to activate and configure the thesaurus:

1. Open the menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Thesaurus configuration**
3. Select a view

View: Catalog Product Search ▼

Synonyms configuration



Enable synonyms search

Yes



Use
Default

[CONTAINER
- STORE
VIEW]

Synonyms weight divider

4



Use
Default

[CONTAINER
- STORE
VIEW]

Query expansions configuration



Enable search expansions

Yes



Use
Default

[CONTAINER
- STORE
VIEW]

Concepts weight divider

10



Use
Default

[CONTAINER
- STORE
VIEW]

4. Change the following settings:

Enable synonyms search: enable/disable the thesaurus of synonyms.

Synonyms weight divider: the score of searches using a synonym will be divided by this value.

Enable search expansions: enable/disable the thesaurus of expansions.

Concepts weight divider: the score of searches using an expansion will be divided by this value.



Recommended values

Concepts weight divider: we recommend setting this value to 10. Expansions are mainly used to populate the queue of the search results and to avoid responses without results. For this reason we recommend to set a high value.

Synonyms weight divider: we recommend setting this value between 2 and 10, depending on the words in your thesaurus. Use a low value if each pair of synonyms has a very similar meaning; use a high value if the meaning is slightly different.

11.3.b Synonyms

Synonyms are set up in the back-office:

1. Open the menu **Smile ElasticSuite > Thesaurus**
2. Click on **Add new thesaurus**
3. Select **Synonym** in **Thesaurus Type**

General Information

Thesaurus Name *

Store *

All Store Views

Main Website

Main Website Store

Default Store View

 ?

Column	Synonym terms	Action
	<input type="text" value="blue,denim"/>	
	<input type="text" value="rose,fuchsia"/>	

4. Change the following settings:

Thesaurus name: Label used to identify this list of synonyms, for back-office use only. This name will not be used by the search engine.

Store: Select the store(s) view(s) where this list should be applied. You can select several store views by using the CTRL key of your keyboard.

Synonyms terms: Type in the column **Synonym terms** a list of words with a similar meaning with comma as a separator. For each new group of synonyms click on the **Add** button.

5. Click on **Save Thesaurus**



Information

When a thesaurus is updated in the back-office, the search engine applies the change after about 1 minute.

11.3.c Expansions

Expansions are set up in the back-office:

1. Open the menu **Smile ElasticSuite > Thesaurus**
2. Click on **Add new thesaurus**
3. Select **Expansion** in **Thesaurus Type**

General Information

Thesaurus Name *

Store *

All Store Views

Main Website

Main Website Store

Default Store View

 ?

Column	Reference Term	Expansion terms	Action
	<input type="text" value="top"/>	<input type="text" value="blouse,bodysuit,shirt,sweatshirt,t-shirt,tur"/>	
	<input type="text" value="bottom"/>	<input type="text" value="pant,short,skirt"/>	
	<input type="text" value="sport"/>	<input type="text" value="fitness,jogging,running"/>	
<div>Add</div>			

4. Change the following settings:

Thesaurus name: Label used to identify this list of expansions, for back-office use only. This name will not be used by the search engine.

Store: Select the store(s) view(s) where this list should be applied. You can select several store views by using the CTRL key of your keyboard.

Reference term: Enter a word that may be expanded into several other words. Only one word is allowed.

Expansion terms: List of words that may be used for a search on the reference term. Use a comma as a separator.

5. Click on **Save Thesaurus**



Information

When a thesaurus is updated in the back-office, the search engine applies the change after about 1 minute.

II.4 AUTOMATIC STOPWORDS DETECTION

Feature added
in version 2.0.0

When the user enters a query with several words, the short words like « and, or, to, for, are, be, by, in,... » are most of the time very frequently found in the catalog, so the short words might have a strong weight in the search result despite their lack of relevance. Most search engines on the market respond to this problem by using a list of « stop words ». The search engine will ignore any word in this list.

ElasticSuite responds to this problem in a different way: the search engine calculates the frequency of each word in the catalog, and ignores the words with a frequency above a given limit, called « cutoff frequency ». If all words in the request are above the cutoff frequency, then the query is searched as a normal query.

This feature allows a more accurate and automatic detection of the meaningless words. And it also allows to process queries with only stop words, like « to be or not to be » for example.

The cutoff frequency can be set up in the back-office:

1. Open the menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Relevance configuration > Cutoff frequency configuration**

Cutoff frequency	<input type="text" value="0.15"/>	[CONTAINER - STORE VIEW]
A number between 0 and 1. Used as automatic stopwords detection threshold.		

3. Enter a value between 0 and 1 in the setting **Cutoff frequency**. Any word included in more than this percentage of indexed entities will be ignored by the search engine.



Recommended values

A usual value for the cutoff frequency is 0.15.

0.15 means that a word will not be significant for the search engine if this word is found in more than 15% of the indexed entities. An indexed entity is a product or a category.

II.5 PHRASE MATCHING

Feature added
in version 2.0.0

The « phrase matching » recognize words that should be used together.



Example

If a user enters the query « gone with the wind », then the search engine may produce bad results if he displays products including any of the words « gone », « with », « the » and « wind ». It will be much better to display a product including the whole sentence « gone with the wind ».

ElasticSuite provides this feature natively.

The phrase matching is working by searching all the sub-groups of words into the query of the user. For each sub-group, a query is run to find the products including the words of the sub-group in the same order relative to each other. Then the results of all sub-groups are combined together, with a boost on the sub-groups including a higher number of words of the query of the user.



Example

If a user enters the query "Dress Giorgio Armani", then the search engine will search for the following queries: (the symbol "..." can replace any group of letters)

Step 1.

Search for "Dress...Giorgio...Armani"

Step 2.

Search for "Dress...Giorgio"

Search for "Giorgio...Armani"

Search for "Dress...Armani"

Step 3.

Search for "Dress"

Search for "Giorgio"

Search for "Armani"

When these 7 queries has been run, all the results are combined together, with a high boost on the products provided by the step 1, a lower boost on products provided by the step 2, and no boost at all on the products provided by the step 3.

The phrase matching can be set up in the back-office.

1. Open in the menu Smile **ElasticSuite > Search Relevance**
2. Select a view

View: Catalog Product Search ▼

3. Open the tab Relevance **Configuration > Fulltext base settings**

Enable boost on phrase match	<input type="text" value="Yes"/>	▼	[CONTAINER - STORE VIEW]
Phrase match boost value	<input type="text" value="4"/>		[CONTAINER - STORE VIEW]

4. Change the following settings:

Enable boost on phrase match: enable/disable the phrase match boost

Phrase match boost value: multiplier to apply to the score of the search engine when a group of words is found



Recommended values

We recommend enabling the phrase match boost.
A usual value for the **Phrase match boost value** is 4.

11.6 MINIMUM NUMBER OF WORDS MATCHING THE QUERY

*Feature added
in version 2.0.0*

When a user enters a query with several words, several approaches could be used:

- | Display the products that includes ALL the words of the query
- or
- | Display the products that includes AT LAST one word of the query

The first approach is better for relevancy, but may return no result for queries with a lot of words.

The second approach will return more results, but the results will be less relevant.

ElasticSuite allows you to select one of these approaches, or to select a compromise between both:

1. Open the menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Relevance Configuration > Fulltext base settings**
3. The setting **Minimum Should Match** allows you to choose the minimum percentage of words that should match in the user query in order to suggest a result. Stopwords are not taken into account when calculating this percentage.

Minimum Should Match



Example

If the **Minimum Should Match** is set to 60%, then:

- | For a query with 3 words, a product will be suggested if at least 2 words are found in the datas of the product.
- | For a query with 5 words, a product will be suggested if at least 3 words are found in datas of the product.



Recommended values

We recommend using a value of 100% (all words should match).
Optionally, a slightly lower value may improve the results for long queries.

11.7 TIE BREAKER: COMBINATION OF SEVERAL PRODUCT ATTRIBUTES

Feature added
in version 2.0.0

The search engine sort products by a relevancy score. This score of a product is calculated in the following way:

- Step 1.** A relevance score is calculated for each attribute: name, description, SKU, brand, color...
- Step 2.** Each attribute score is multiplied by the search weight of the attribute.
- Step 3.** Resulting scores are combined together into a global score.

For the step 3, several methods of combination can be selected:

- | Sum of all scores
- | Maximum score
- | Maximum score + (sum of other scores) * fixed_percentage

To change the calculation method:

1. Open then menu **Smile ElasticSuite > Search Relevance**
2. Open the tab **Relevance Configuration > Fulltext base settings**
3. The setting **Tie breaker** should be set to one of the following values:

Tie breaker

1: Select the method « Sum of all scores ».

0: Select the method « Maximum score ».

Any value between 0 and 1: select the third method, and the value entered will be used as the fixed_percentage.



Recommended values

We recommend using a value of 1.

A lower value may be interesting when several attributes with a similar weight include redundant keywords.



11.8 BETTER AUTOCOMPLETE

Feature added
in version 2.1.0

In native Magento, the autocomplete suggests only popular searches (similar queries from other customers).

With ElasticSuite the autocomplete also suggests products, and with a quick development a developer can easily add other contents: static pages, shops, lookbooks...

Products suggestions are using all the powerful features of the search engine: search weight, fuzzy search, phonetic search, automatic stop words detection...

pant		Q
Search terms		
pant		37
pant red		14
pant for yoga		7
Products		
	Cronus Yoga Pant	\$48.00
	Karmen Yoga Pant	\$39.00

Autocomplete can be set up in the back-office:

1. Open the menu **Stores > Configuration**
2. Open the tab **ElasticSuite > Autocomplete**
3. Change the maximum number of results that can be displayed using the settings **Max size**.

Popular Term Autocomplete

Max size [STORE VIEW]

Maximum number of popular search terms to display in autocomplete results.

Product Autocomplete

Max size [STORE VIEW]

Maximum number of products to display in autocomplete results.

Then, you have to select the products attributes that should be used to suggest products:



4. Open the menu **Store > Attributes > Product**
5. Select a product attribute
6. Change the following settings:

Used in autocomplete

Display in autocomplete

Used in autocomplete: If set to **Yes**, then this attribute will be used for product suggestions. For example, if this parameter is set to Yes on the product attribute « color », then when a user enters « yellow » the autocomplete will suggest yellow products.

Display in autocomplete: This setting is not used at today (this feature is in the roadmap).

yellow		Q
Products		
	Diva Gym Tee	\$32.00
	Tiberius Gym Tank	\$18.00

III - FILTERS

III.1 MULTIPLE SELECT IN LAYERED NAVIGATION

Feature added
in version 2.0.0

With the native search engine of Magento, the customer is able to combine several filters in the same time.



Example

Material = Cotton AND Color = Black

ElasticSuite enhances this feature by adding the ability to select multiple values in the same filter.



Example

Material = Cotton OR Organic Cotton

Multiples selections can be combined with other filters, like in Magento native search engine.

- MATERIAL** ▼
- ☐ Polyester (22)
 - ☒ Cotton (16)
 - ☐ Spandex (13)
 - ☐ Nylon (10)
 - ☒ Organic Cotton (9)
 - ☐ CoolTech™ (6)
 - ☐ Fleece (6)
 - ☐ Lycra® (6)



Example

(Material = Cotton OR Organic Cotton) AND (Color = Black)

III.2 ABILITY TO SORT VALUES IN FILTERS

Feature added
in version 2.0.0

The sorting order of the values in a filter can be customized for each filter:

1. Open the menu **Stores > Attributes > Product**
2. Select an attribute that can be used as a filter
3. Open the tab **Storefront Properties**
4. Change the value of **Facet sort order**

Facet sort
order

Result count ▼

Result count: The values shared by the most products will be displayed first.

Admin sort: Manual sorting order. The sorting order is managed in the tab **Properties**.

Name: Alphabetical order.

Relevance: The values shared by the products in the top of the list will be displayed first.

ACTIVITY ▼

- ☐ Gym (15)
- ☐ Recreation (9)
- ☐ Sports (8)
- ☐ Athletic (7)
- ☐ Yoga (7)
- ☐ Outdoor (4)

Values sorted by Result count

5. Option: you can change the maximum number of values displayed in the filter by using the setting **Facet max. size**.

Facet max.
size

10

Max number of values returned by a facet query.

III.3 ENHANCED PRICE SLIDER

Feature added
in version 2.0.0

ElasticSuite displays the price filter as a slider instead of fixed price bands, providing more flexibility on the selection of the price range.

PRICE ^

€ 20,00 - € 29,99 (4)

€ 30,00 - € 39,99 (7)

€ 40,00 and above (1)

Native Magento

PRICE ▼

\$22.00 \$78.00

25 products

OK

With ElasticSuite

III.4 SUPPORT OF LARGE CATALOGS

Feature added
in version 2.0.0

On large catalogs, a simple search query may return dozens of filters, too much filters to be displayed entirely. As a result, some important filters may not be displayed.

ElasticSuite provides a solution to display only the filters that are the most relevant for the query of the user. This can be achieved by setting up the minimum percentage of products that should share an attribute in order to display the attribute in the list of filters. This percentage is called the **Facet coverage rate**.



Example

The search query « Black » returns 1000 products. A lot of filters can be applied to that query:

- ! Price (attribute shared by 1000 products)

- | Color (attribute shared by 900 products)
- | Size (attribute shared by 400 products)
- | Battery Capacity (attribute shared by 30 products)
- | Etc.

After setting up a « minimum coverage rate » of 20%, ElasticSuite will display the filters Price, Color, and Size (shared by more than 20% of the products), and will **not** display the Battery Capacity filter (shared by less than 20% of the products).

This Minimum Coverage Rate can be set up in the back-office:

1. Open the menu **Stores > Attributes > Product**
2. Select an attribute that can be used as a filter
3. Open the tab **Storefront Properties**
4. Change the value of the **Facet coverage rate**

Facet coverage rate	90
Ex: Brand facet will be displayed only if 90% of the product have a brand.	



Recommended values

The value of the coverage rate should be adjusted depending on two factors:

- (1) the percentage of products using this attribute in the catalog
- (2) the number of filters in the whole catalog

For example, if the Battery Capacity is used by 5% of the products (1), then we can assume that this filter is relevant only if more than 5% of products in the search results are sharing this attribute. Thus the Facet Coverage Rate should be set to a value higher than 5.

It is recommended to start by setting up the Facet Coverage Rate to the value of the percentage (1) multiplied by a ratio of 2. When this work has been done on all filters, it is recommended to run several queries and look at how many filters are displayed. Depending on the results, the ratio can be adjusted to a higher or to a lower value.

IV - MERCHANDISING

IV.1 VIRTUAL CATEGORIES

Feature added
in version 2.1.0

One of the key factor of the success of an e-commerce website is the ability to help customers to:

- | **Find** quickly what they are looking for
- | **Discover** all the range of products

A way to cover that need is to provide several axis of navigation in addition to the « main navigation » by department.

The screenshot shows a navigation menu for the 'KIDS' department. The main navigation bar includes 'KIDS', 'SHOES & ACCESSORIES', 'LINGERIE', 'NIGHTWEAR', and 'SALE'. Below 'KIDS', there are three main sections: 'BOYS BY AGE', 'GIRLS BY AGE', and 'COLLECTIONS'. 'BOYS BY AGE' includes 'Baby Boys (0 - 24 Months)', 'Younger Boys (2 - 6 Years)', and 'Teen Boys (7 - 14 Years)'. 'GIRLS BY AGE' includes 'Baby Girls (0 - 24 Months)', 'Younger Girls (2 - 6 Years)', and 'Teen Girls (7 - 14 Years)'. Below these, there are 'BOYS' and 'GIRLS' sections with 'View All' and 'New Arrivals' links, followed by a list of sub-categories like 'Coats & Jackets', 'Knitwear', 'Tops', 'Jeans', 'Trousers & Shorts', 'Babygrows & All In Ones', 'Nightwear & Slippers', 'Underwear & Socks', 'Shoes', 'Accessories', 'Sets & Outfits', 'Swimwear', and 'Basics'. The 'COLLECTIONS' section lists 'Holiday Shop', 'Character Shop', 'Disney', 'Frozen', 'Minions', 'Peppa Pig', 'Minecraft', 'Star Wars', 'Marvel', 'Spiderman', 'Girls Fancy Dress', 'Boys Fancy Dress', and 'Kids Schoolwear'. There are also 'KIDS OFFERS' (e.g., '£3 Off Schoolwear'), 'GIFT CARD', and 'E-GIFT CARD' sections. A legend on the right indicates that orange boxes represent the 'Main category tree' and green boxes represent 'Secondary category trees'.

BOYS BY AGE	GIRLS BY AGE	COLLECTIONS
Baby Boys (0 - 24 Months)	Baby Girls (0 - 24 Months)	Holiday Shop
Younger Boys (2 - 6 Years)	Younger Girls (2 - 6 Years)	Character Shop
Teen Boys (7 - 14 Years)	Teen Girls (7 - 14 Years)	Disney
		Frozen
		Minions
		Peppa Pig
		Minecraft
		Star Wars
		Marvel
		Spiderman
		Girls Fancy Dress
		Boys Fancy Dress
		Kids Schoolwear

BOYS
View All
New Arrivals
Coats & Jackets
Knitwear
Tops
Jeans
Trousers & Shorts
Babygrows & All In Ones
Nightwear & Slippers
Underwear & Socks
Shoes
Accessories
Sets & Outfits
Swimwear
Basics

GIRLS
View All
New Arrivals
Coats & Jackets
Dresses & Skirts
Knitwear
Tops
Jeans
Leggings & Jeggings
Trousers & Shorts
Babygrows & All In Ones
Nightwear & Slippers
Underwear & Socks
Shoes
Accessories
Sets & Outfits
Sportswear
Swimwear
Basics

KIDS OFFERS
£3 Off Schoolwear

GIFT CARD

E-GIFT CARD

Main category tree
Secondary category trees

Example of use

Secondary categories can be created using manual selections of products, for example:

- | **Shop by look**
- | **Gift ideas**

With ElasticSuite, secondary categories can also be created using an automatic selection of products based on a rule. When a rule has been set up, the list of products displayed to the customer **is updated in real-time**.



Note

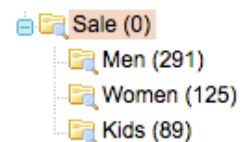
With native Magento, it's possible to create a category based on a rule, but the rule engine is not so advanced, and the rule is run only one time: when the category is created. The list of product **is not** updated in real-time.

Examples of categories based on a rule:

- | **Sale** : all discounted products
- | **Brand page** : all products from a given brand
- | **New products / New arrivals**
- | **Good deals / Great offers** : all products with a price below a given limit
- | **Shop by collection**, for example « Winter 2017 » : all products with the attribute « Collection = Winter 2017 »
- | **Shop by material**, for example « Linen selection » : all products with the attribute « Material = linen »
- | **Shop by age**, for example « Teen boys » : all products with the attribute « Age = 7-14 years » and « Gender = Men »
- | **Shop by size**



A virtual category can also have sub-categories, created manually, or automatically by ElasticSuite.



To create a new virtual category:

1. Open the menu **Products > Categories**

2. Create a new category or select an existing category
3. Open the tab **Merchandising**


4. Set **Yes** in **Virtual category**

Virtual category



5. Enter the rule of the virtual category.


Virtual rule

If ALL of these conditions are TRUE :

New is Yes 

Category is one of

  ☐ Default Category (2046)



You can set-up a rule based on product datas using a meta-language. If a product attribute is missing, you can add it from the menu **Stores > Attributes > Product** by setting **Yes** in the parameter **Use for Promo Rules Condition**.

6. Option: if you want to display a facet « Category » in your virtual category, then you have to choose the category tree of this facet in the block **Virtual category root**

Virtual category root

Not Selected

IV.2 SORTING PRODUCTS BY DRAG & DROP

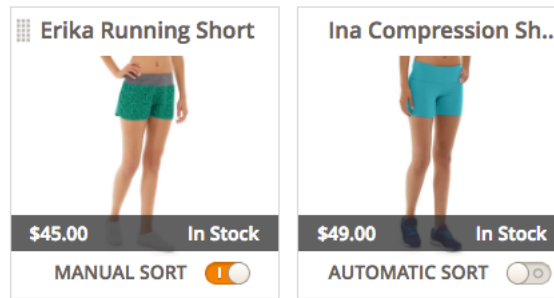
*Feature added
in version 2.1.0*

In Magento Enterprise, a feature enables the merchant to sort manually the products in a category by drag & drop. But this feature has two strong limitations:

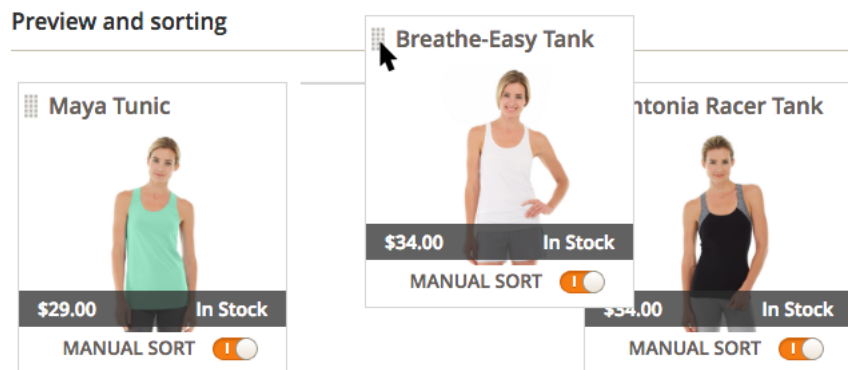
- 1 The merchant must sort each size and each color, involving a lot of operations to get the expected result.
- 2 The merchant is able to sort the products only in the categories at the bottom of the category tree; it is not possible to sort the products on the parent categories (categories with « anchor = yes »).

ElasticSuite solves these two limitations and adds the support of sorting by drag & drop in virtual categories:

1. Open the menu **Products > Categories**
2. Select a category
3. Open the **Merchandising** tab
4. Enable the **Manual sort** on several products. Products sorted manually are automatically moved to the top of the page.



5. Click on the handle in the top left corner and drag & drop the products.



6. Save the category.

V - TECHNICAL IMPROVEMENTS

ElasticSuite is based on the search engine Elastic Search, providing several technical enhancements to the native Magento search engine.

| High Performances

Built on top of Apache Lucene, ElasticSearch delivers high performances allowing you to perform complex queries on large catalogs.

| Massively Distributed

Elasticsearch allows you to start with a small hardware and add more nodes later ; the cluster will automatically take advantage of the extra hardware.

| High Availability

The replication feature detect new or failed nodes, and reorganize and rebalance data automatically, to ensure that your data is safe and accessible.

| Per-Operation Persistence

Document changes are recorded in transaction logs on multiple nodes in the cluster to minimize the chance of any data loss.