

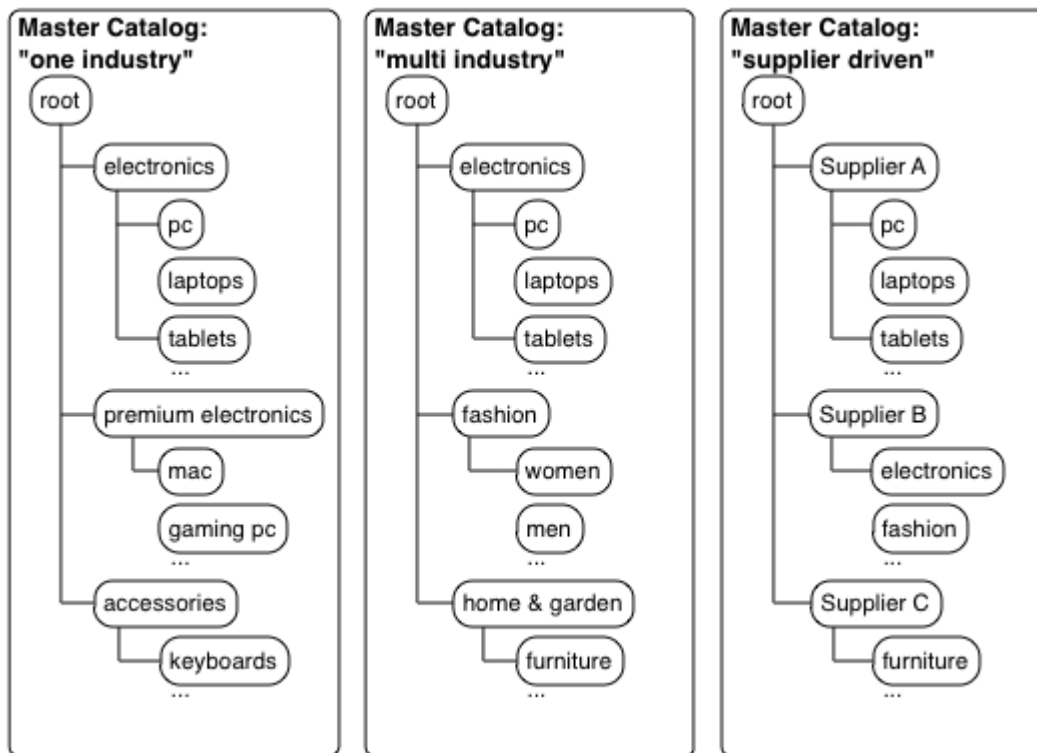
# Catalog

- Overview
- Category management
  - Managing master catalog hierarchy
  - Managing category
    - Main
    - Localization
    - Templates
    - SEO
    - Search Index
    - Attributes

## Overview

Catalog management refers to management of product categories which is a separate topic to [product information management \(PIM\)](#). Best way to think about **catalog management is classification of products**, where categories act as groups for products with some defining features. For example if we consider fashion products we may encounter top level classifications such as "Men", "Women" and "Kids" with each one further refined. Women category could contain "dresses", "tops" and so on and each one of these can be further subclassed to fulfil business requirements. Classification does not necessarily have to represent a feature of the products contained within it. For example category "sale" can represent a mix of products that need to be classified as items on sale. Thus a product can be classified as say "Women / Dress" but also as a "Sale" item. **Products in the platform can have multiple category assignments** thus it is possible to list (or rather classify) product under several categories.

If we expand on the idea of classification it becomes clear that catalogs are in essence just branches of a classification hierarchy. This is the approach that the platform takes - there is only one so to speak "master" catalog that holds all branches of category classifications. Thus if the business operates on a variety of products types it makes sense to make top level classifications by the nature of products, say top level categories could be: "Electronics", "Fashion" and "Home & Garden", where each can be further refined as required. If we take this to yet another level of complexity and say that we categorise in the first place by "supplier catalog", it could makes sense to set the top level categories as names of suppliers. Visually it can be depicted below with "one industry", "multi industry" and "supplier driven" catalog layouts.



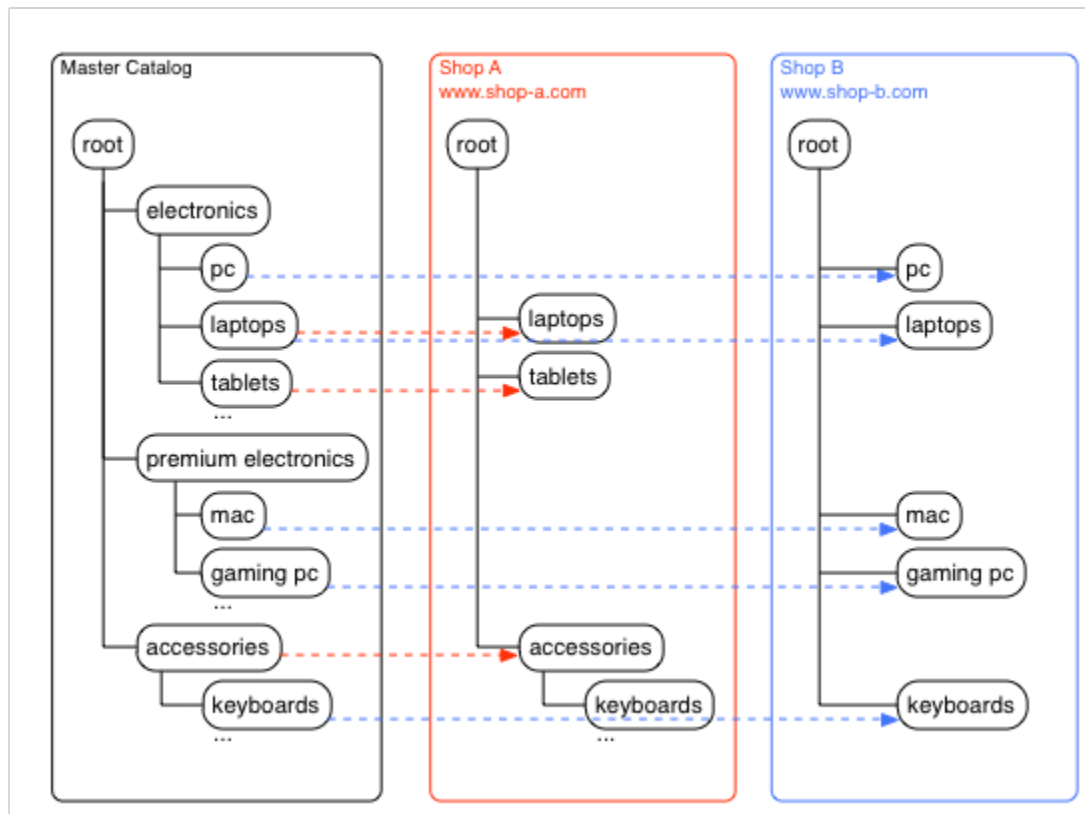
Essentially the platform allows building hierarchy as your business requires so that categories can be used as classification of products that the business sells. This classification is not however necessarily what the customers on the website sees, because the "shop catalog" (i.e. customer facing catalog) is defined by shop catalog assignments.

When [shops](#) are being configured categories from master catalog are picked to become top level categories in the shop. If chosen category has sub categories the shop that assigns it will automatically inherit that branch of hierarchy. Thus it is possible to "pick & mix" various branches from the master catalog which then collectively become shop specific catalog. If you have multiple shops and one sells only electronics then third level categories "pc", "laptops" and "tablets" can be directly assigned thus making only that part of master catalog available in that specific shop. Say another shop sells "Electronics" and "Fashion" only then second level category can be assigned (excluding "Home & Garden" products). More complex example when a shop sells "PC" from "Supplier A" and "Fashion" products from "Supplier B".

Concept of sharing master catalog in such a way works particularly well for multi-store, multi-brand and marketplace businesses but just as well in simpler single shop installations.

Master catalog hierarchy and the way it ties with shops is heavily dependent on business requirements. It is highly recommended to analyse business processes to determine the most appropriate catalog structure.

#### Basic multi-store example

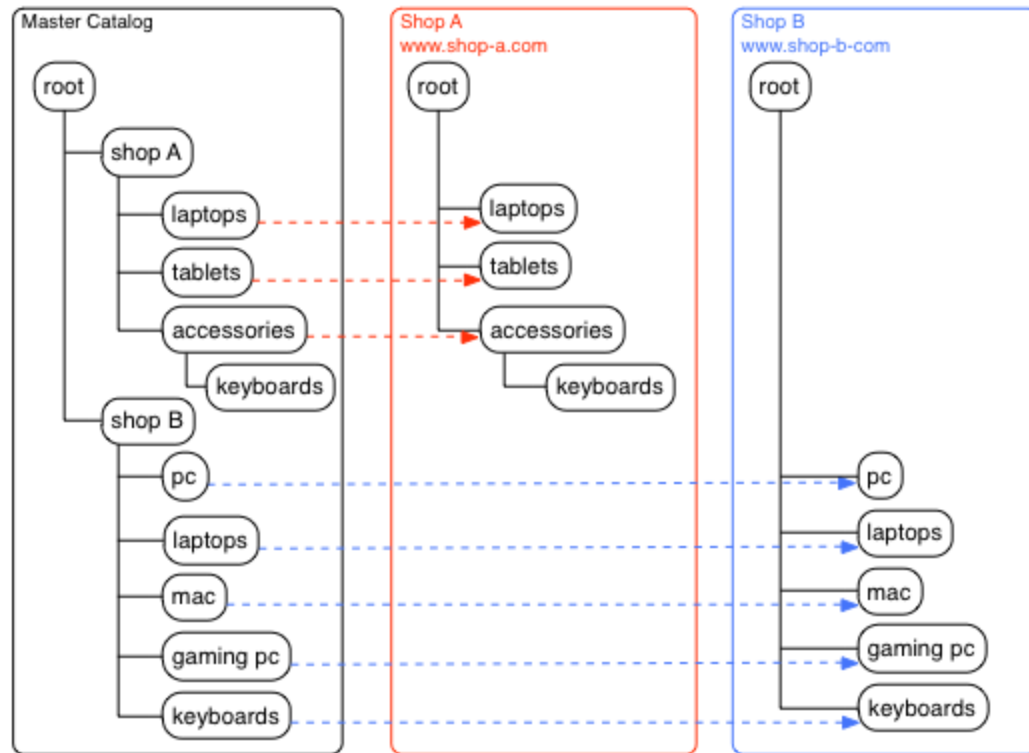


Multi-store example master catalog's categories are assigned to "shop A" and "shop B" at different levels thus providing each shop with its own unique category structure. Note that there is no limitation on how categories are assigned to shops. For example "shop A" assigns category "accessories" and thus inherits "keyboards" subdirectory whereas "shop B" only assigns "keyboards" category directly. Category "laptops" is assigned to both "shop A" and "shop B". Any permutation of assignments is allowed.

Benefits of this approach is that once products are added to any category they become available to all shops provided those shops have [stock](#) and define a selling [price](#) for products contained in those categories.

The drawback is that master catalog has to be carefully managed to provide flexible category hierarchy to suit all shops that share it. In most cases this is achievable since most product categorisation is fairly standardised. However if using common hierarchy is not feasible then alternative category structure of master catalog can be proposed where master catalog has separate hierarchies per shop.

#### Per shop hierarchy multi-store example

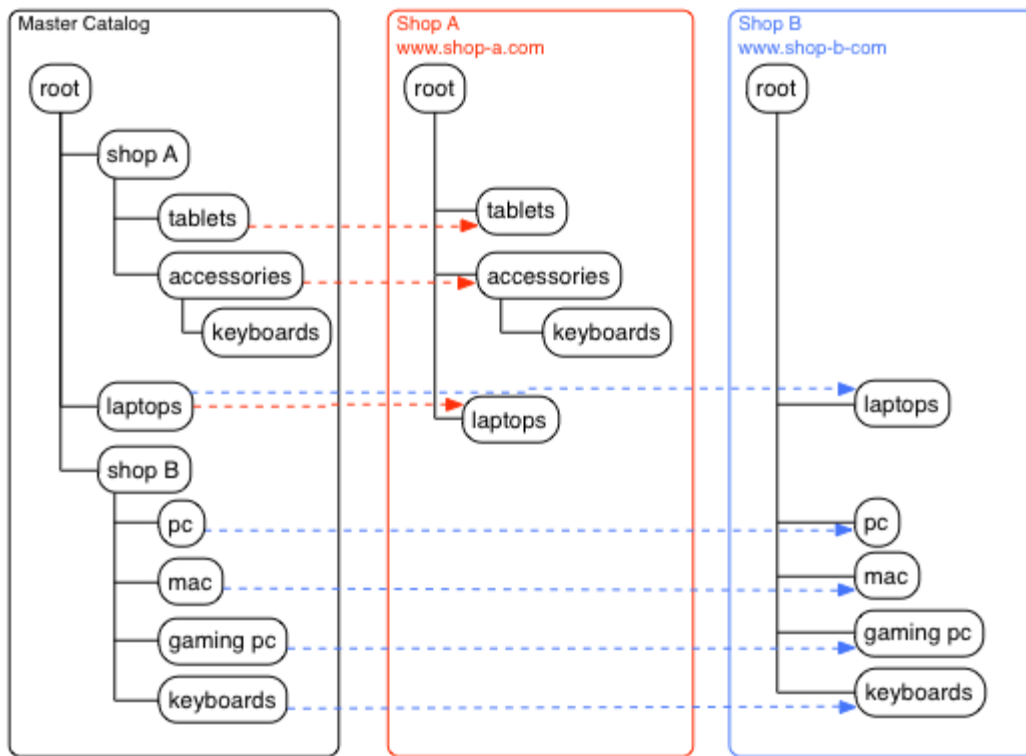


With per-shop configuration master catalog has categories for each shop with sub categories that reflect each shop's category structure. This approach allows complete freedom of crafting unique catalog structures at the shop level.

There are two drawbacks of such approach. First is the duplication of categories. For example category "laptops" had to be declared twice since both "shop A" and "shop B" have it. Second is the necessity for products that belong to the "laptop" category to be assigned to both categories whereas previously only one assignment was required. Of course in some scenarios this may be seen as a benefit where two laptops categories actually contain different products. Once again it depends on the business requirements.

It is possible to mix two approaches described above.

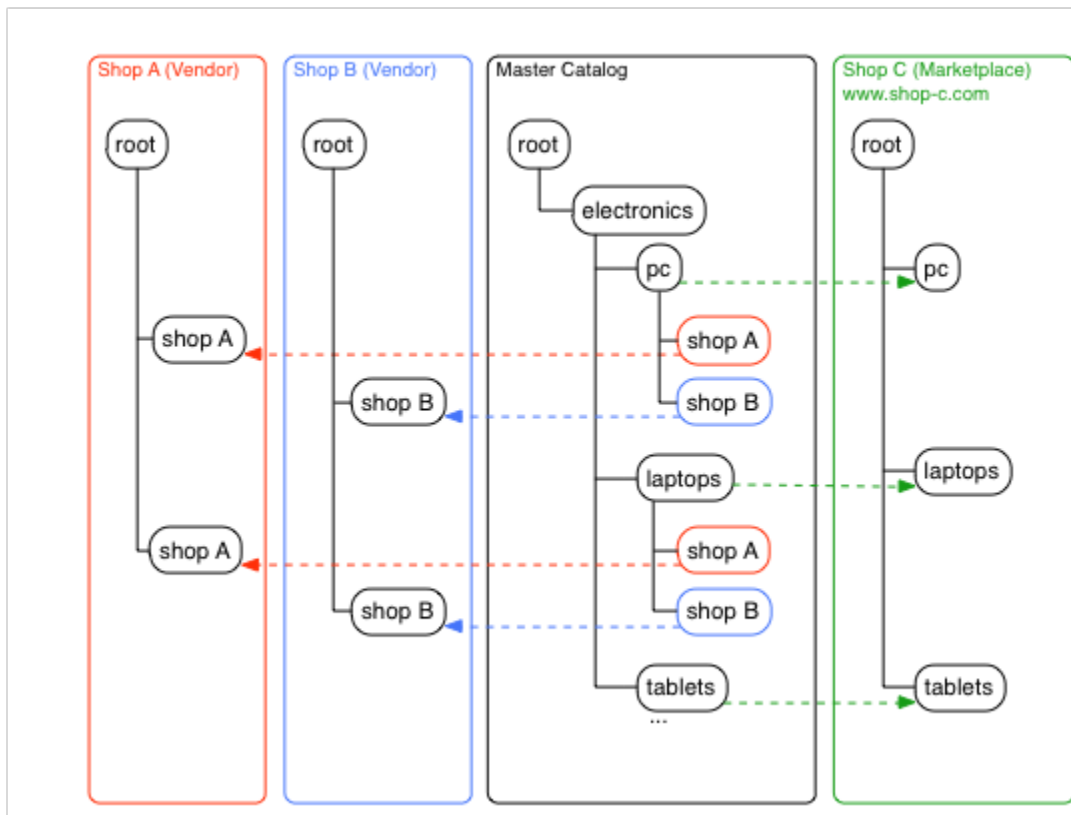
### **Mixed approach**



In mixed approach shop specific categories are defined in shop specific hierarchies in the master catalog. However "laptops" category that is shared by both shops is positioned one level up and assigned to both shops.

Another advanced example of catalog structure organisation is for marketplace installations.

#### Marketplace example



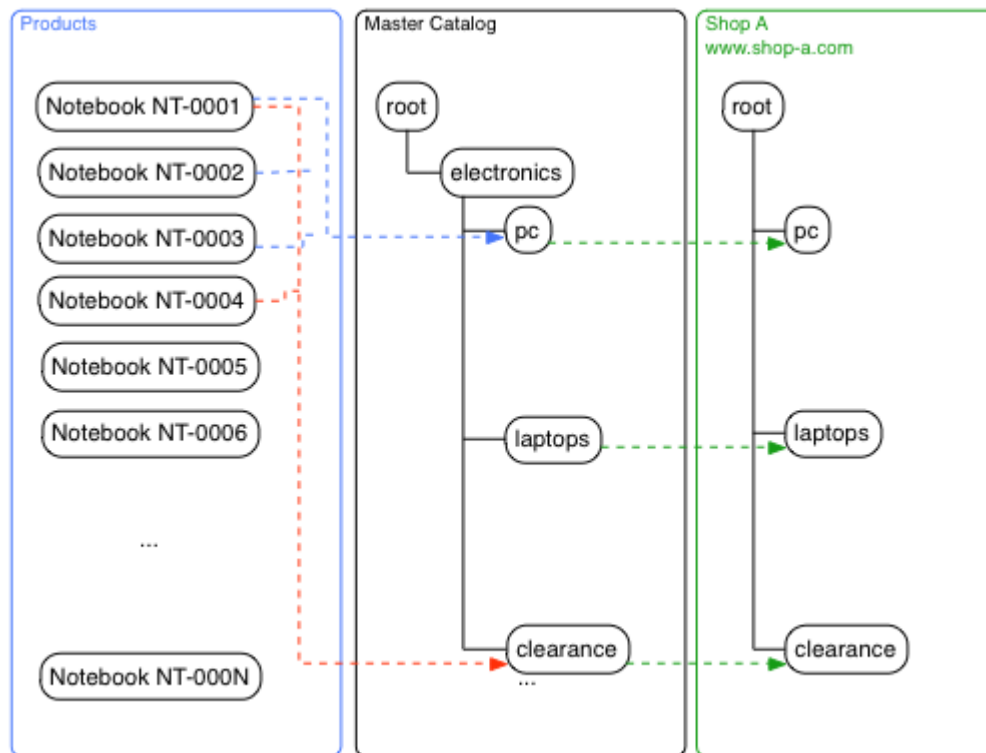
In this example we have three shop instances: "shop A" and "shop B" are vendors which do not have customer facing URL mapping and "shop C", which is the marketplace that consolidates products from all vendors.

With this setup master catalog hierarchy is configured in accordance to category structure of "shop C". However each category declares linking categories to vendors. This allows all products loaded by vendors into categories "shop A" and "shop B" to be available in "pc" category as well. All "shop A" and "shop B" categories need to be assigned to "shop A" and "shop B" shop instances respectively, so that vendors will have access to master catalog and be able to [import](#) products and use [PIM](#) section.

As can be seen from examples above the master catalog mechanism is very flexible and category hierarchy organisation strategy is heavily dependent on the business requirements. Hence we would like to stress the point that **analysing your business and choosing the right strategy for master catalog is very important**.

Once the category structure approach is selected product categorisation has to be carefully considered as well. The platform provides **many-to-many relationship between products and categories** meaning that a product can be assigned to one or more categories. This gives extra level of flexibility when plain products cataloging does not fully fulfil the requirements.

Consider example where a separate "clearance" category is required. Assigning product to its original category and "clearance" category makes it available in both categories on the website without the need to duplicate product data.



In example above "Notebook NT-0001" is assigned to "pc" and "clearance" categories and will be available in both. "Notebook NT-0002" and "Notebook NT-0003" only assigned to "pc" and "Notebook NT-0004" only to "clearance". Therefore it possible to move product from category to category as well as have it in multiple categories.

Product category assignment provides additional level of flexibility in respect to "product to shop" association. Therefore business users may take advantage of this mechanism in order to share products between shops and providing same product in multiple categories on the website.

Some examples of these concepts can be seen in [search and navigation](#) categories section

## Category management

### Managing master catalog hierarchy

Catalog management panel allows to manipulate master catalog structure with ease through adding and removing categories. There are two options to create a category: **add** using the add button, or **link** which will create a linking category to target one that is selected. Linking categories is very powerful mechanism whereby you can create an alternative set of top level categories for shop which links through to designated branches. This is an alternative approach to "catalog per shop" strategy described in overview section.

Default view of the categories is data table as it is easier to work with when searching for categories and viewing basic data available in table column. However because categories structure is hierarchical there is also a **hierarchy button** which allows to give "bird's eye" view of the master catalog and navigate to specific branch.

To add new category to hierarchy whether a simple category or a linked one a parent category has to be selected. Parent can be chosen by selecting it from hierarchical view. Category can be moved to any other branch by simply changing the parent.

The screenshot displays the Catalog Management interface. At the top, there's a header with "DEMO ENVIRONMENT" and "Logged in as Yes". Below the header, a "Catalog" section shows a search bar with "YCDemo" and a "Current filter" label. A table lists categories with columns: Code, Name / URI, Category template, Navigation, Available from, Available to, and Available. The table includes categories like "YC Demo Catalog (500)", "Laptops (500)", "PCs/Workstations (500)", etc. A "Selected Activity" label points to the table. A "Summary of filtered navigation (facets)" label points to the "Navigation" column. A "Hierarchy view" modal is open, showing a tree structure of categories. A "Category editor" modal is also open, showing details for the "Laptops" category, including its parent, code, rank, and availability dates.

Code	Name / URI	Category template	Navigation	Available from	Available to	Available
YCDemo	YC Demo Catalog (500)					✓
151	Laptops (500)		\$ Laptops			✓
153	PCs/Workstations (500)		\$ PCs/Workstations			✓
156	Servers (500)		\$ Servers			✓
194	Keyboards & Desktops (500)		\$ Key			✓
195	Mice (500)		\$ Mice			✓
375	Toners & Laser Cartridges (500)		\$ Ton			✓
788	Warranty & Support Extensions (500)		\$ War			✓
803	Software Licenses/Upgrades (500)		\$ Sof			✓
897	Tablets (500)		\$ Tab			✓

**Category editor**

Parent: root > YC Demo Catalog

Code: 151

Rank: 500

Available from: yyyy-MM-dd HH:mm:ss

Available to: yyyy-MM-dd HH:mm:ss

There is **no limit on how many sub category levels there are**. It is up to business to decide how granular the master catalog categorisation should be.

In order to remove the category "delete" button can be used. However only categories that do not have product assignments and sub categories can be removed. It is recommended to simply make category unavailable to front end by setting "disabled" flag or "available to" date to past value instead of deleting it.

Once the master catalog hierarchy is established categories can be assigned to [shops](#).

## Managing category

Category editor contains a number of tabs providing a focused view of the category data.

### Main

Summary screen contains the basic category data.

**Parent** field that defines the position of given category in hierarchy. Note that the parent can be changed thus "moving" category to a different location in hierarchy and keeping all settings.

**Code** is used by [import](#) and is optional, since if left blank it will be auto generated. For catalogs that involve manual management (i.e. not imported from third party systems) It is recommended to create codes manually as this will make referencing categories easier in catalog and product import files.

**Rank** is used for manual sorting of the categories. Default sorting is alphabetical which is used in rendering customer facing menus. Use ranking to create manual sort order. If rank of two categories is the same then alphabetical sorting is applied. To keep alphabetical sorting of sub categories set same rank for all sub categories (or leave it as default which will set to 500)

**Disabled, Available from/to** control visibility of the category on storefront. If category is disabled by flag or current date does not fall into range of from/todates **category cannot be navigated to on the website, all products assigned to it will still be available on the website (i.e. via global search)**.

### Localization

Contains client facing category **name** which is used in rendering category menus and a **description**.

### Templates

Templates refers to template that should be used to render this category's page on the website.

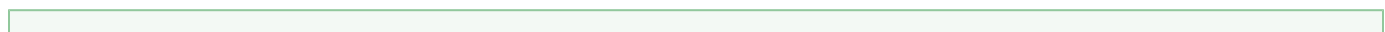
Default behaviour is displaying products if category has products assignments (note that if category does not have direct assignments but sub categories do and search in subcategories flag is enabled then products view is rendered), otherwise sub categories view is display if there are any, else category view is rendered.

More formally the rules are:

1. if template is specified use it; otherwise
2. if there are products available and [product type search template](#) is configured use it; otherwise
3. if there are products available then choose product listing rendering; otherwise
4. if there are sub categories then choose category listing; otherwise
5. simple category rendering.

You can override this mechanism by providing specific template to use by given category (i.e. using rule #1). The following template keys can be used out of the box from the [default storefront theme](#).

Template	Version	Description
products	3.0.0+	Default behaviour (category with products), display products listed in this category
category	3.0.0+	Default behaviour, display category description in the page's main body
subcats	3.0.0+	Default behaviour (category with sub categories), display category description in the page's main body
catcontent	YCE 3.4.0 +	Uses category description field as placeholder for content URI. Body of this content is displayed in page's main body



Custom themes can define more templates that can provide alternative behaviours

## SEO

**SEO** tab provides settings for values such as **title** and **meta tags** and **URI**. URI refers to last part of the URL to category page. By default all categories map to the following pattern:

```
http://www.shopdomain.com/category/[URI]
```

Therefore if value is set to "tablet-pc" for example the final SEO friendly URL leading to this category would be:


```
http://www.shopdomain.com/category/tablet-pc
```

If URI is not specified then internal category ID is used.

## Search Index

Search index tab contains filtered navigation (or facets) configuration.

The following configurations are available:

Configuration	Version	Description
brands	up to 3.4.0	flag enables brands filtered navigation block. Brand values are accumulated from products that are assigned to given category (and its sub categories if "search in sub categories" flag is enabled).  <b>Note that in versions 3.4.0+ brand value is "proxied" by brand attribute so that it can be added to generate attributes filter navigation block and thus use some of the features such as sorting and alternative naming</b>
price		flag + per currency price range settings enable price filtered navigation block. The configurations are set using "price tiers edit" button. The tiers have to be set for each currency separately. This is not limitation of the platform but rather best practices approach. Business marketing department has to set correct ranges for product prices that will be balanced and will be suitable to target customer groups. For example if there are many products grouped with middle tier prices it would make sense to create "up to X" price range, then make granular ranges for middle range prices and then last range to have "more than X". Another reason is there are sometime "gaps" in price ranges thus creating 0 counts, so marketing department may choose to group this range with adjacent one to prevent 0 value ranges. Sometimes it makes sense to create different ranges based currency since currency may refer to geo location customer group with different price perception due to local mentality. Price range counts are accumulated from products prices that are assigned to given category (and its sub categories if "search in sub categories" flag is enabled).
attributes		flag enables attribute filtered navigation blocks that require default category <a href="#">product type</a> to determine which attributes should be part of the navigation. <b>For each attribute a separate navigation block will be created</b> subject to sorting and navigation rules defined by selected product type. Attribute values are accumulated from products that are assigned to given category (and its sub categories if "search in sub categories" flag is enabled).

## Attributes

As stated in the [attributes management](#) section attributes allow to add additional data to category object. Some of these represent flags and configurations used on the storefront.

Attribute	Version	Purpose
Category Description (multiple attributes)	3.0.0+	Allows to set long descriptions that are language specific. In <a href="#">default theme</a> this is used by simple category rendering template as body of the page



Quantity of featured items	3.0.0+	Number of products in featured item component. In <a href="#">default theme</a> this is used by category featured products template
Quantity of new arrival items	3.0.0+	Number of products in new arrivals tab. In <a href="#">default theme</a> this is used by category new arrival tab template
New arrival tag days offset	3.0.0+	Number of days in the past for product to be identified as new arrival product if it does not have "newarrival" tag. In <a href="#">default theme</a> this is used by category new arrival tab template to determine which products to show
Items per page CSV	3.0.0+	Pagination settings for category with products. In <a href="#">default theme</a> this is used by product lister category template to display options for number of products to be displayed on a single page
Category sort options CSV	3.0.0+	Product sorting settings for category with products. In <a href="#">default theme</a> this is used by product lister category template to display options for products sorting for listing or search result
Search include sub categories	3.0.0+	Flag that allows to alter <a href="#">search engine behaviour</a> . If this is set to "on" then listing or searching products in this category will also include products from sub categories.
Quantity of category pods in one row on category page	3.1.0+	Quantity of product pods in one row to show on category page
Quantity of category pods in one row on category page	3.1.0+	Quantity of category pods in one row to show on category page
Filter navigation records limit	3.2.0+	Filter navigation records limit per group. Default is 25. If there are less records than limit they are alphabetically sorted, otherwise sorted by most hits descending and over the limit are discarded
Flag to exclude this category from meganav	SaaS 3.3.0+	Used by SaaS default theme to skip this category when rendering mega nav