Product information (PIM)

- Overview
- Key concepts for storefront
- Detailed documentation

Overview

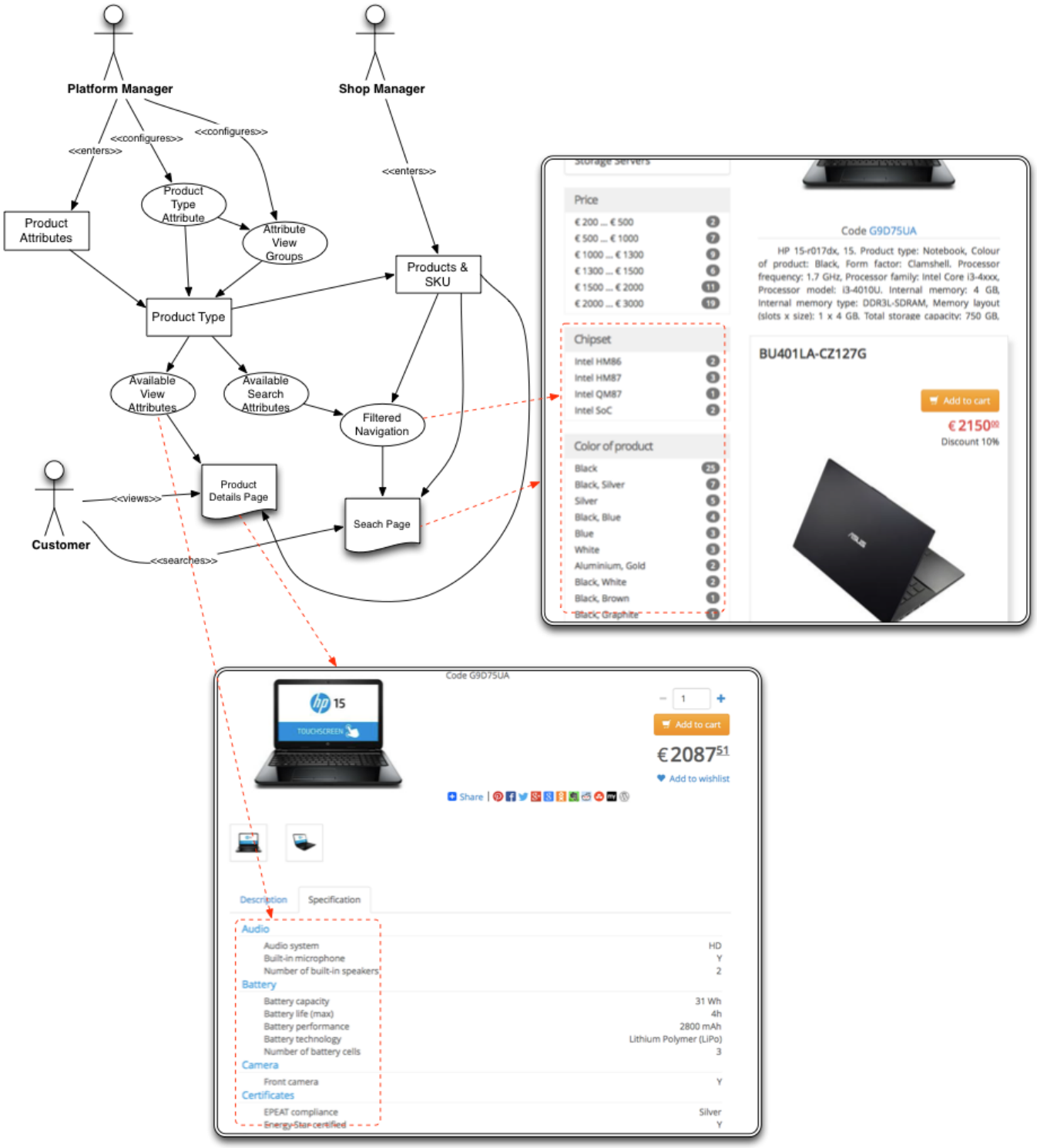
The foundation of PIM is provision of *configurable product types*. Although products share some core information that allows them to be processed by e-commerce systems (such as SKU code and price) this information is far from being enough from the customers' point of view. Customer require detailed information about what they are buying and once we are getting into this realm we quickly realise that all products are very different. Consider the specification parameters for a TV versus groceries versus beauty products versus fashion products. Each of these product types require different set of data to describe them efficiently to the customers and encourage them to proceed with the purchase.

In order to solve this problem the platform offers a concept of **product types**, which are configurations to provide a *focused view on the product information*. Product types allow to:

- Create attribute **view groups** to define product **parameters relating to a common concept**. For example "Audio" view group defined for "Laptop" product type can consist of "Audio system", "Built-in microphone", "Number of built-in speakers" which would give a complete view on the audio capabilities of a particular laptop.
- Create **searchable attributes**. Most manufacturer provide very detailed product information, which works very well when there is a limited set of products. Once we get into thousands of products much of the information starts to overlap and introduce a lot of "noise", which makes searching for specific products difficult. This is especially true for attributes such as product descriptions. For example consider a phrase "this will look nice with our red hand bag" for a "dress" specified in products's description, which would make this "dress" a part of "hand bag" phrase search or "red" colour. This is not necessary wrong (in some use cases) but in general can leave customers puzzled or even annoyed to see dresses in their searches for hand bags or a dress in a different colour. This is is why the platform allows to specify exactly, which attributes should **contribute to search** thus providing **only clean and relevant keywords**.
- Create **navigatable attributes**. Some parameters are very important to customers purchase decision making process and thus we put emphasis by configuring them to be part of the navigation menu. For example "CPU speed" or "laptop screen diagonal" could be much more influential factors than "number of USB ports" for a "Notebook" product. In order to place those into attribute navigation all is needed is to select a checkbox in product type attribute configuration.
- Provide **comparable views** SaaS . Once the product type views are configured it is trivial to present information to the user showing key differences by listing comparable parameters side by side.

Product information management through **product types configuration can be fully managed via Admin** without any need for custom code. Depiction of the approach can be seen in the figure below:

Figure 1: Product type and product data attributes data organisation



Once the product types are configured in the system, admin users can simply import product data or manage it via editors within Admin app.

> For automated product imports it is possible to configure import service as a recurring import job thus fully automating process of updating product information for businesses with existing external PIM systems.

Key concepts for storefront

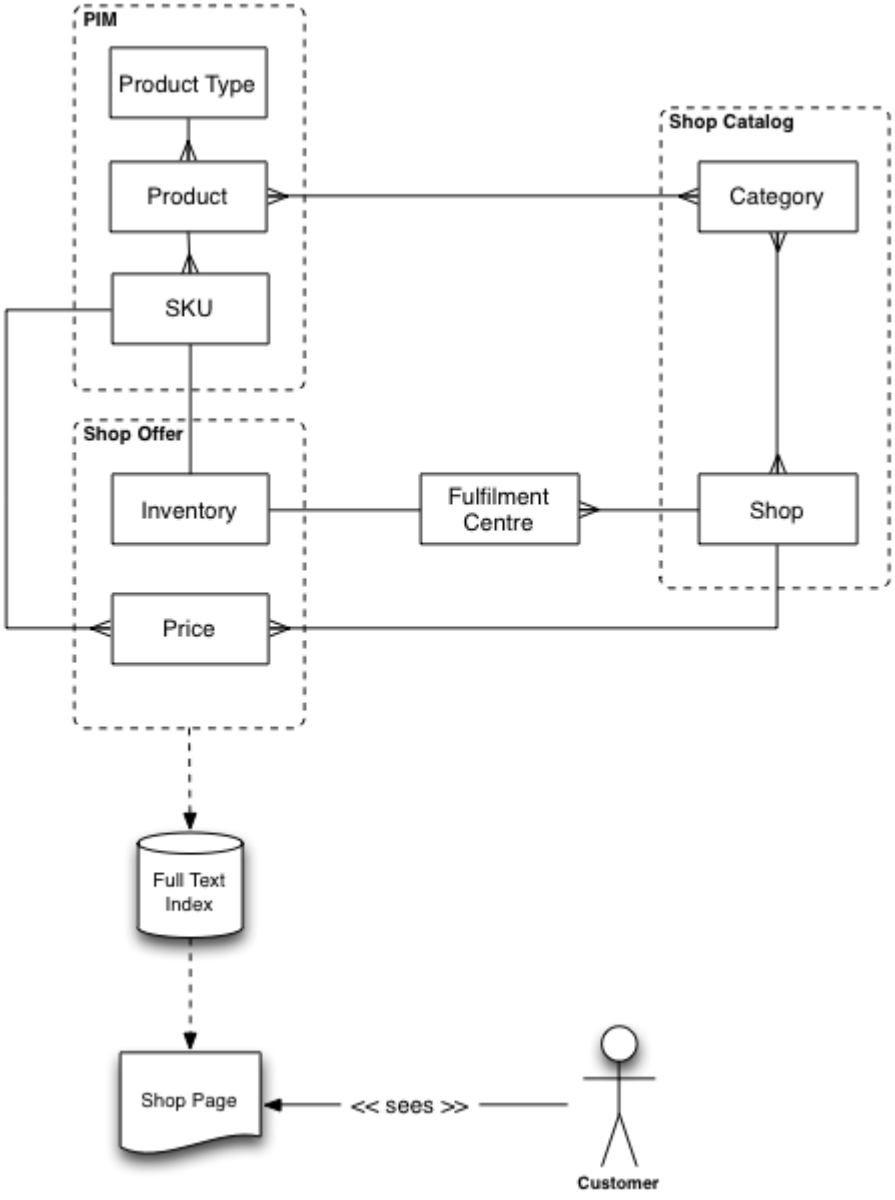
PIM is platform global and is therefore **shared by all shops** hosted on the platform. PIM in terms of the platform is comprised of the following data:

- product type definition
- product definition containing basic information and all custom attributes
- SKU definitions containing basic information and all specific custom attributes or overrides from inherited product custom attributes

in essence it is a repository of product information which is generic and can apply to any reseller who wishes to sell these product.

Therefore in **order to be displayed to the customer products has to be offered by the shop**, which in terms of platform means:

- **SKU has to have inventory record** in fulfilment centre available to the shop (i.e. this shop has this item in its inventory)
- **SKU has to have at least one price record** defined for the shop (i.e. this shop offers this product at a given price)
- **Product has to be available in a shop category** (i.e. this shop can display this product from its category menu)
- If all of the above holds true **after product index is build** the products will appear in frontend.



As depicted in the diagram:

- **PIM** defines **how the product looks**,
- **Shop Catalog** defines **which products can be navigated to** and
- **Shop Offer** defines **inventory and prices**.

Full text search index holds all this **data combined** together **is scope of a shop** and makes it available the customer to see.

Detailed documentation

> It is highly recommended to start with product type configuration overview to gain understanding into how product information in made available to the storefront and which part thereof affect specific functional areas.

- IceCat integration
- Product data
- Product types