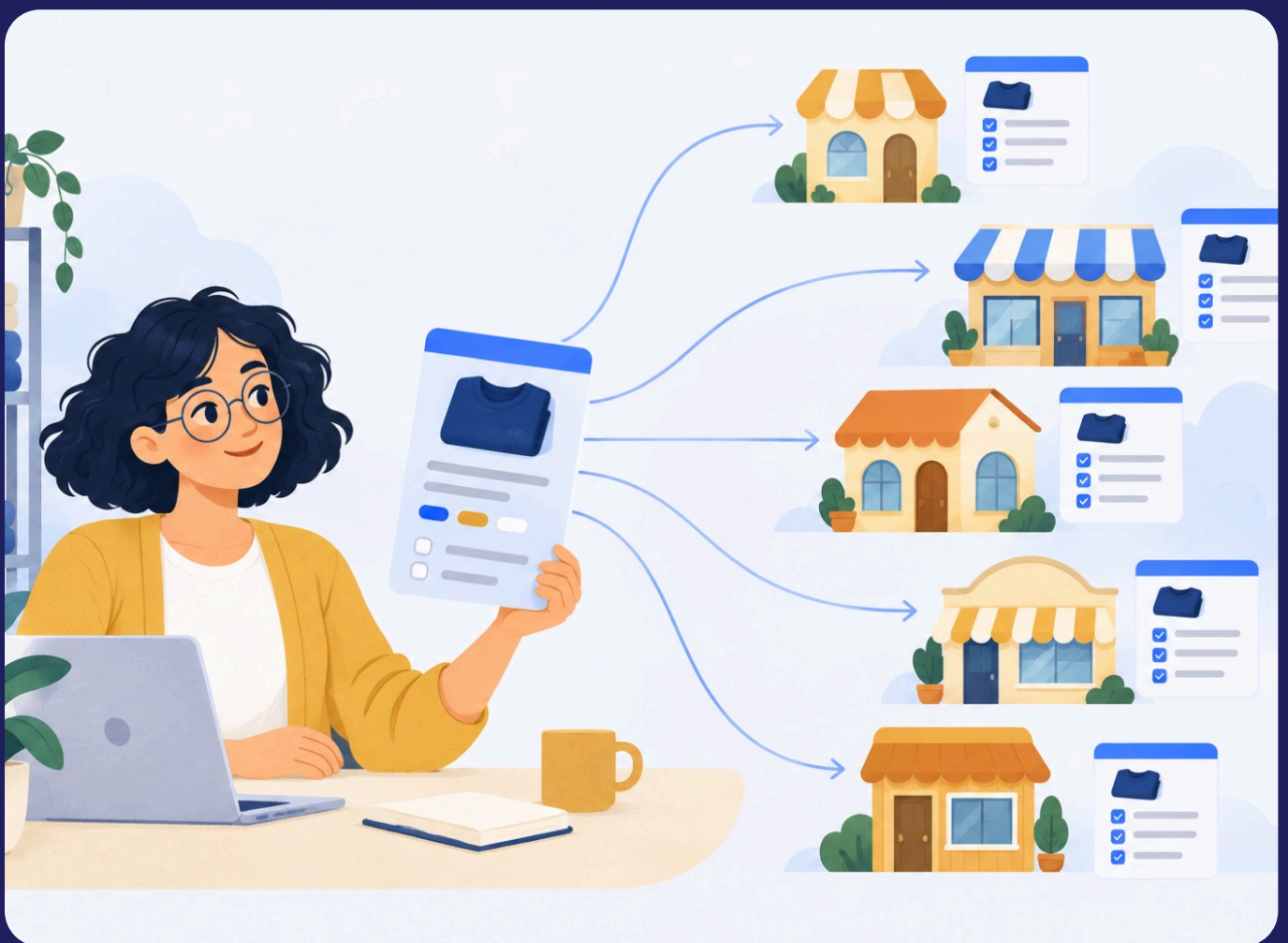


Product rules

A product rule defines what a kind of product is: which properties belong together, and how strict each one is. It brings structure to your catalogue, and tells you when a product is finished.



Done this before? Try the questions on **page 11**. All of them right? Go straight to the next module.

What is a product rule?

A T-shirt and a sofa need different details. The **product type** says what kind of product something is. Its **product rule** groups the properties that belong to that kind of product, and says how strict each one is. Together they define what a T-shirt is in your catalogue.

T-Shirt	Product rule	Sofa	Product rule
Colour	Required in Configurator	Colour	Required
Size	Required in Configurator	Width (cm)	Required
Brand	Required	Depth (cm)	Required
Material	Required	Seats	Required
Fit	Optional	Cover Material	Optional

Colour appears in both rules. It's the same property from module 1: you describe colour once, and every rule that needs it uses it.

What rules are for

Defining what a product is

Grouping properties turns a pile of loose details into "a T-shirt" or "a sofa".

Structure

Every product of a type follows the same list, in the same order. Even where there was little structure before.

Consistency

Whoever adds a product, it's described like the others of its kind. Shoppers see the same details on each, and can compare.

The right fields

A product only shows the properties its rule lists. No sofa width on a T-shirt.

Knowing when it's finished

Empty properties are shown as missing. Empty Required ones also block the product from going to your channels.

Your channels

Rules are instructions for your channels. On your online store they become the structure. Marketplaces check your products against their own. Module 3.

Rules bring structure

Without rules, every product is described its own way. With them, each kind of product has one clear shape. That brings consistency and predictability, for you and for your customers. You know what every T-shirt has, and OneSila checks it. Shoppers find the same details on every T-shirt, so they can compare and filter.

WHY IT MATTERS

One T-shirt has a fit and no material. The next has a collar style and no size. A third has a brand nobody else uses. Nobody can say what a T-shirt in your catalogue is supposed to look like, so nobody can say which ones are finished. And a shopper comparing two of your T-shirts finds different details on each.

In this module you learn to

1. Explain what a product rule does.
2. Choose how broad a product type should be.
3. Read a rule: what's required, what's optional, what the customer chooses.
4. Decide what's Required in a rule.
5. Know which products a rule change affects, and what moving a product to another type does.

Elsewhere, something similar

WHERE, THEIR WORD	HOW IT COMPARES TO A PRODUCT RULE
Magento Attribute set	The closest match: a set of attributes for one kind of product. Each OneSila rule becomes an attribute set on Magento (module 3).
Akeneo and other PIM systems Family	Close: the attributes a kind of product has, and which ones are required.
Amazon Product type	Amazon's own version: its list of what a kind of product needs. You can't change it; you match yours to it (module 3).
Shopify Product type	Only a label on the product. No list of fields comes with it.
eBay Category	Where an item is listed. eBay expects certain details per category, but a category isn't a list you define.

How broad is a product type?

A product type groups products that are described the same way. Draw the line too wide or too narrow, and you pay for it later. The goal is a balance.

TOO BROAD

Toys

Teddy bears, board games and dolls in one type. The rule can't say what a toy needs: a board game has no fabric, a teddy bear has no number of players. Almost everything ends up Optional, so nothing blocks an unfinished toy. Marketplaces also split finer than "Toys", which makes matching harder later.

TOO NARROW

Green Teddy Bears

Then Blue Teddy Bears, Red Teddy Bears... The same list, many times over. Every change has to be made in each rule, and each one matched to every channel separately.

ABOUT RIGHT

Soft Toy

Teddy bears, plush rabbits, any colour. They share one list of properties and the same checks. Colour is a property of the soft toy, not a type of its own.

A RULE OF THUMB

Do these products need the same list of properties? Then they can share a product type. Does the list differ a lot? Split them. A difference in colour or size is a value, never a new product type.



Ask: what would I check on every one of these? If the answer is the same list for all of them, it's one product type.



Ask: where will I sell them? If a marketplace treats them as clearly different kinds of product, your types probably should too.

Read a rule

On the left is every property you have. On the right, the ones this product type needs. To add one, click the plus in front of its name, or drag it from the left column to the right. OneSila then asks for its **rule type**: how strict it is.

Need a property that doesn't exist yet? The round blue plus above the left column creates a new one, in a new tab. The round arrows next to it refresh the list. To remove a property from the rule, click its bin or drag it back to the left. Drag within the right column to change the order.

Product Rules > Edit Product Rule

Product Type: T-Shirt (Edit name) Sales Channel: Default

Require Ean Code: ☒

Search: [] [+] [↻] Search: [] Property Type: [v] Rule Type: [v]

Name	Property Type	Name	Property Type	Rule Type	Actions
+ Accent Color	Short Text	Colour	Single Select	Required in Configurator	↓
+ Accents	Multiselect	Size	Single Select	Required in Configurator	↑ ↓
+ Accessories	Single Select	Brand	Single Select	Required	↑ ↓
+ Additional Parts Required	Single Select	Designed In	Single Select	Optional	↑ ↓
+ Amazon Merchant Shipping Group	Single Select	Material	Single Select	Required	↑ ↓
+ Amazon Offering Can Be Gift Messaged	Yes/No Choice	Fit	Single Select	Optional	↑ ↓
+ Amazon Product Tax Code	Single Select	Fabric weight (gr/m2)	Whole Number	Required	↑ ↓
+ Amazon Special Features	Single Select	Age Gender Category	Description	Optional	↑ ↓
+ Animal Theme	Single Select				
+ Apparel Fabric Stretch	Single Select				
+ Apparel Fabric Weight Class	Single Select				

Product Rules > Edit Product Rule, T-Shirt. Colour and Size are *Required in Configurator*, Brand and Material are *Required*, Fit is *Optional*. **Sales Channel: Default** means this is the Default rule: it applies everywhere. A channel can get its own version (module 3).

MUST HAVE A VALUE

NICE TO HAVE

A DETAIL

Required

The product isn't finished until this has a value.

Optional

Shown on the product, but nobody has to fill it in.

THE CUSTOMER CHOOSES IT

Required in Configurator

The choice a shopper makes when buying, like size. Also required.

Optional in Configurator

Only offered as a choice when the versions really differ.

STEP 3

Fill in a product

Each product's **Properties** tab shows exactly the properties from its rule, in the same order. You only fill in the values.

The screenshot shows the 'Properties' tab for a product named 'Orange Organic Cotton T-Shirt'. At the top, there are four status indicators: 'Required' (0), 'Optional' (0), 'Done' (10), and 'Suggested' (0). A 'Suggest values' button is visible in the top right. Below the status indicators is a search bar and a dropdown menu for 'Type' (set to 'All') and a language selector (set to 'English'). The main area displays a list of properties for the 'T-Shirt' product type. Each property has a status indicator (Required, Optional, Done, Suggested) and a value field. The properties are: Colour (Red), Size (Medium), Brand (Girld Dad), Designed In (Italy), Material (100% Organic Cotton), Fit (Classic), Fabric weight (gr/m2) (180), and Age Gender Category (Gender category information). Each property field has a dropdown arrow, a plus icon, and a trash icon.

Orange Organic Cotton T-Shirt › Properties. Its product type is T-Shirt, so the T-Shirt rule applies. Every property has a value, so Done shows 10. Top right: **Suggest values**.

Required

How many required properties still have no value. This is the number to get to zero.

Optional

How many optional properties are still empty. Nice to fill, not a blocker.

Done

How many properties already have a value.

LET AI FILL THE GAPS: SUGGEST VALUES

Click **Suggest values** on the Properties tab. OneSila's AI reads the product's name, description and photos, and what its other versions already have, then proposes values for the empty lists and Yes/No properties. It only picks values you already have, so the clean values from module 1 pay off here. Each suggestion waits in its field: accept it with the check, or dismiss it. Nothing changes until you accept. It uses AI credits.

The product type sits at the top of the Properties tab. Change it, and the product follows another rule: other properties, other checks.

What the customer chooses

Many T-shirts come in several sizes and colours. Only the size and colour change; everything else stays the same. So you don't make 15 separate T-shirts. You make one, with versions.

Balboa Persistence Tee
 SKU: MS03
 Type: Configurable Product
 Created At: 10/04/2025, 15:56
 Active: ✔
 Workflows: +

Missing Optional information Default ↺

1 Product has undecided sales channel views

Duplicate Delete

Variations

Name	SKU	Configurator Value	Product Type	Active	Content Quality	Actions
Balboa Persistence Tee-XS-Gray	MS03-XS-Gray	XS x Grey	Top	✔	80%	Delete
Balboa Persistence Tee-XS-Green	MS03-XS-Green	XS x Green	Top	✔	80%	Delete
Balboa Persistence Tee-XS-Orange	MS03-XS-Orange	XS x Orange	Top	✔	80%	Delete
Balboa Persistence Tee-S-Gray	MS03-S-Gray	S x Grey	Top	✔	80%	Delete
Balboa Persistence Tee-S-Green	MS03-S-Green	S x Green	Top	✔	80%	Delete
Balboa Persistence Tee-S-Orange	MS03-S-Orange	S x Orange	Top	✔	80%	Delete

Balboa Persistence Tee › Variations. One product holding its versions. The Configurator Value column shows what tells them apart: XS x Grey, S x Green.

The parent holds the versions

Think of the parent as a folder. Each version inside is a full product with its own SKU, following the same rule. The properties set to **Required in Configurator** tell the versions apart.

Not every product has versions

A T-shirt that comes in one size and one colour has no versions: it's a Simple Product. There, Required in Configurator simply acts as Required.

THE WORDS IN ONESILA

A product on its own is a **Simple Product**. A parent holding versions is a **Configurable Product**, and the versions are its **Variations**. What tells them apart must be a Single Select: one clear choice, like "M". Module 1 explains why.

Decide what's Required

With everything Optional, gaps still show, but nothing blocks. Required properties do block: until they have a value, the product isn't sent to your storefront or marketplaces.

Product Rules › Edit Product Rule, Top. A rule where every property is Optional: gaps are shown, but nothing blocks. Imported rules start like this (see the course Working with channels).

What to switch to Required

- 1 What every product of this type must have.** If one is missing it, is the product really finished? Then it's Required.
- 2 What shoppers filter on.** A product without a size disappears from the size filter.
- 3 What the customer chooses.** Size, colour: switch those to Required in Configurator.

NOT EVERYTHING, THOUGH

Everything else can stay Optional: it's still shown on the product, it just doesn't block anything. The usual mistake goes the other way: marking every property Required, to be safe. Then a product missing one minor detail counts as unfinished, and unfinished products aren't sent to your channels. Keep Required for what really matters.

THEN LET AI FILL THE GAPS

Made something Required, and now many products have gaps? Use **Suggest values** (step 3). It proposes values from what OneSila already knows about each product, and you accept or dismiss each one.

Changing a rule

A rule isn't a setting for one product. Any change will change the rules for every product of that type. On a rule with 678 products, that's 678 products, not 1.

0 selected of 24			
☐	Product Type	Usage count	Actions
☐	Dog Carrier	8	Edit Delete
☐	Sofa with Mattress	1	Edit Delete
☐	T-Shirt	2	Edit Delete
☐	Duffle Bag	2	Edit Delete
☐	Bag	18	Edit Delete
☐	Bottom	226	Edit Delete
☐	Default	0	Edit Delete
☐	Downloadable	0	Edit Delete
☐	Gear	19	Edit Delete
☐	Sprite Stasis Ball	9	Edit Delete
☐	Sprite Yoga Strap	3	Edit Delete
☐	Top	678	Edit Delete

Properties › Product Rules. Usage count shows how many products follow each rule: 678 products are a Top.

A REAL CASE

A customer changed a rule each time one product got stuck. Changing a rule isn't wrong in itself. A new channel or a growing catalogue often needs it. But every change applied to all products of that type, and products that were finished suddenly weren't.

Before you change a rule, check how many products follow it and what the change does to them. If you find yourself changing a rule for one product at a time, the product type may be too broad (step 1).

OneSila asks first. When products use the rule, saving shows **Confirm product rule change** with the number of products, and you type the rule's name to approve. Read that number.

CHANNELS CONNECTED?

Then a rule change reaches them too. Your online store shows it, and marketplaces check your products again. Module 3 explains it.

Moving a product to another type

A product's type decides which rule it follows. Move it to another type, and it follows the other rule. Anything the new rule doesn't list isn't sent with the product anymore.

A REAL CASE

A customer moved products to another product type. The new rule was missing properties those products had, and their website lost that information. The fix was one of two: move the products back to their old type, or add the missing properties to the new rule.

Before you move a product

- 1 Compare the two rules.** What does the product have now that the new rule doesn't list?
- 2 Add what's missing to the new rule first.** That rule is shared too. If other products of that type don't need it, add it as Optional.
- 3 Move one product, check it, then the rest.**

What a move does on your live channels: module 3, and the course [Working with channels](#).

YOUR TURN

Open **Properties › Product Rules** and pick a product type with many products. Open its rule and look at the Rule Type column. Is everything Optional? Write down the properties every product of that type should have. Those are your candidates for Required.

Six questions

Pick one answer for each. The answers, and why, are on the next page.

1 "T-Shirt", and the list of properties every T-shirt needs. Which is which?

- A. T-Shirt is the rule, the list is the product type
- B. T-Shirt is the product type, the list is its product rule
- C. Both are product rules
- D. Both are properties

2 What does a product rule do?

- A. Defines what a kind of product is: which properties belong together, how strict each is, and so what goes to your channels
- B. Only checks whether a product is finished
- C. Holds each product's values, like Navy or M
- D. Lists the options a property can have, like S, M and L

3 You sell teddy bears in many colours, dolls and board games. Which product types make most sense?

- A. One type for all of them: Toys
- B. One type per colour of teddy bear
- C. One type per kind of product described the same way: Soft Toy, Doll, Board Game
- D. No product types at all

4 The customer picks a size when buying. Which rule type fits Size?

- A. Optional
- B. Required
- C. Yes/No Choice
- D. Required in Configurator

5 You remove Fit from the T-Shirt rule. 678 products follow that rule. Which products change?

- A. Only the T-shirt you have open
- B. Only T-shirts you add from now on
- C. All 678
- D. None, until you confirm each product

6 A colleague wants to make every property in the T-Shirt rule Required, to be safe. Good idea?

- A. No. Products missing one minor detail count as unfinished, and aren't sent to your channels
- B. Yes, it's the safest choice
- C. Yes, Suggest values will fill the gaps
- D. Only for rules with few products

Answers, and why

B**Question 1 · B. T-Shirt is the product type, the list is its product rule**

The product type says what kind of product it is. Its rule lists the properties that kind of product needs.

A**Question 2 · A. Defines what a kind of product is**

It groups the properties that belong together, and says how strict each one is. That list is also what goes to your storefront and marketplaces. Knowing when a product is finished (B) follows from that, but it isn't all a rule does. The values themselves (C) sit on each product. The options of a property (D) are its Select Values, from module 1.

C**Question 3 · C. One type per kind of product described the same way**

"Toys" (A) is too broad: a board game and a teddy bear need different details. One type per colour (B) repeats the same rule over and over: colour is a value, not a product type.

D**Question 4 · D. Required in Configurator**

It's the choice the customer makes, and every version needs one. Yes/No Choice is a property type, not a rule type.

C**Question 5 · C. All 678**

The rule is shared by every product of its type, so a change reaches all of them. Before you save, OneSila shows that number. Read it.

A**Question 6 · A. No. Unfinished products aren't sent to your channels**

Required means "not finished without it". Make everything Required and almost nothing is finished. Keep Required for what every product really must have. Suggest values (C) helps fill gaps, but can't know everything.

NEXT

Module 3 · Where your product rules end up: what your rules do on your online store, and on marketplaces.



ONESILA

Define what each kind of product is.

onesila.com