

TRUEBUNDLE · USER MANUAL

TrueBundle User Manual

Mix-and-match bundles with exact pricing and real component inventory, on Shopify's native checkout.

Contents

Welcome to TrueBundle	3
Before you start	3
How a bundle is put together	3
A five-minute setup	3
Creating a bundle	3
The title and parent	4
Pricing your bundle	4
Pools (choose N of M)	4
Component list prices	4
Preflight checks	4
Live preview	5
How pricing stays exact	5
How inventory stays correct	5
Draft and publish	5
Show the picker on your storefront	6
Keeping prices in sync	6
What TrueBundle does not do	6
Troubleshooting	6
Getting help	7

Welcome to TrueBundle

TrueBundle builds mix-and-match bundles that are correct where it counts: the total the shopper pays and the stock you have left.

- **Exact pricing.** The component prices always add up to the total you advertise, to the last cent, in every currency you sell in.
- **Real component inventory.** Each part of a bundle becomes a real order line, so Shopify tracks its stock like any other sale and a bundle can never quietly oversell a component.

It is built entirely on native Shopify surfaces. A bundle is expanded into its real components at checkout by a Shopify Cart Transform Function, and the optional storefront picker is a theme app block. That means TrueBundle never edits your theme files and never leaves you cleaning up after a sale.

This manual walks through setup, building and pricing a bundle, the preflight and preview panels, how the pricing and inventory guarantees work, publishing, the storefront picker, and troubleshooting. If you just want a quick answer, see the FAQ and support page.

Before you start

You need:

- A Shopify store with the TrueBundle app installed.
- A product to sell as the bundle (the parent), and the component products that go inside it.

You do **not** need a developer, theme code changes, or any change to how you already create products.

How a bundle is put together

A TrueBundle bundle has three parts:

- **A parent variant.** This is the product a shopper actually buys. It carries the bundle's configuration and, once published, only sells as the full expanded bundle.
- **Pools.** Each pool is one choice the shopper makes, for example pick any 3 of 8. A pool has a minimum, a maximum, and an optional per item limit.
- **Components.** The product variants inside each pool. At checkout they become the real order lines the shopper receives.

A five-minute setup

1. In your Shopify admin, create a product for the bundle itself. This is the parent the shopper buys.
2. Open TrueBundle and choose **New bundle**.
3. Pick the parent variant, add one or more pools, and add component variants to each pool.
4. Choose how the bundle is priced.
5. Watch the preflight and preview panels confirm the bundle is correct, then **Publish**.

The rest of this manual explains each of these steps in detail.

Creating a bundle

Open TrueBundle and choose **New bundle**, or open an existing one from the list. The builder is one page, from the top: the bundle title and parent, the pricing, the pools, the preflight checks, and a live preview.

The title and parent

- **Title.** An internal name so you can find the bundle in the app's list. Shoppers never see it; on the storefront they see the parent product's own title.
- **Parent variant.** Choose **Select parent variant** and pick the variant shoppers buy. Create the bundle's product first, then pick its variant here. When you publish, this variant is marked as requiring components, so it can only sell as the expanded bundle and never on its own.

Pricing your bundle

Choose one of three pricing modes:

- **Fixed total.** One price for the whole bundle, in your store currency. Checkout converts it once per shopper currency and then splits it across the components so the parts always add up exactly.
- **Percent off the components' list prices.** Enter a percent, up to two decimals, for example 15 or 12.5. The discount comes off the sum of the selected components' list prices.
- **Sum of component prices.** Each component sells at its list price and the bundle total is their sum. Use this when you want a bundle with no discount.

The list prices used for percent-off and for the fixed-total split are the snapshots stored inside the bundle (see "Component list prices" below).

Pools (choose N of M)

Each pool is one choice the shopper makes. Add a pool and set:

- **Minimum picks.** The fewest items a shopper must take from the pool.
- **Maximum picks.** The most they may take.
- **Per item limit (optional).** The most units of any single component. Leave it blank for no limit.

Set the minimum and maximum to the same number for an exact count, for example choose exactly 3. Then choose **Add components** and pick the variants that belong in the pool. You can add as many pools as a bundle needs; each is an independent choice.

Component list prices

When you add a component, TrueBundle snapshots its current price into the bundle. That snapshot is what pricing uses: as the allocation weight for a fixed total, as the basis for a percent off, and as the unit price when the bundle is the sum of its components. You can edit a snapshot in the builder. TrueBundle also refreshes snapshots automatically when you change a product's price later (see "Keeping prices in sync").

Preflight checks

The **Preflight checks** panel runs the same validation the checkout engine runs, live, as you edit. It lists every problem it finds, not just the first, so you can fix them in one pass. Common ones it catches:

- A pool with no components, or a bundle with no pools.
- A minimum greater than the maximum, or a maximum of zero.
- A minimum that can never be met even at the per item limit on every component.
- A list price or a percent that is not a valid amount.
- A bundle where every pool is optional, which could expand to nothing.

You cannot publish until the preflight passes. When it does, the panel shows **Ready to publish**.

Live preview

The **Live preview** panel shows exactly what a shopper will get. It runs the same pricing engine the checkout Cart Transform runs, so it can never show a total checkout would not produce.

- For a fully fixed bundle, the preview expands it automatically.
- For a mix-and-match bundle, enter a sample selection and the preview validates it against the pool rules, then shows the result.
- Set a **Parent line quantity** to confirm that quantities multiply cleanly while unit prices stay the same.

The preview lists each expanded item with its quantity and unit price, and shows the exact total with an **Exact total** badge. Preview prices are shown in your store currency; at checkout the same engine converts once into the shopper's currency and keeps the total exact.

How pricing stays exact

This is the heart of TrueBundle, so it is worth understanding.

- The bundle has one total: a fixed amount, a percent off the components, or the sum of the parts.
- For a fixed total or a percent off, that one total is converted into the shopper's currency **once**, then split across the components. Converting each component separately is how other tools drift by a cent; TrueBundle converts once and then allocates.
- The split uses **largest-remainder allocation**, so the parts always sum back to the total. When a remainder cannot sit evenly on a component's units, that component is split across two lines priced one cent apart. The parts still add up exactly.
- All money is handled in whole minor units (cents), never in floating-point, so there is no rounding surprise.

The result: whatever total you advertise is the total the shopper pays, and the line items always reconcile to it.

How inventory stays correct

TrueBundle expands the bundle into its real components at checkout, so inventory is tracked natively.

- Each selected component becomes a **real order line**, and Shopify decrements that variant's stock like any other sale. There is no separate, drifting count to keep in step.
- The parent variant is flagged to **require components**, so it can only sell as the full expanded bundle. It can never sell "on its own" and leave you owing parts you do not have.
- If a bundle's configuration is ever unreadable or invalid, the parent simply cannot be purchased. TrueBundle fails closed, so a shopper is never sold a bundle at the wrong price or with the wrong parts.

Draft and publish

A bundle is either a **draft** or **active**.

- **Save draft** stores your work in the app without changing anything at checkout.
- **Publish** writes the bundle to Shopify: it registers the component relationships on the parent, flags the parent to require components, and writes the configuration to the private metafield the checkout Function reads. Publishing is done in that order so the configuration only goes live once the parent is ready.

If you edit a bundle that is already active, your changes stay in draft until you publish them. Checkout keeps using the last published configuration until you do, so an in-progress edit never reaches shoppers early.

Show the picker on your storefront

For a mix-and-match bundle, add the **TrueBundle picker** block so shoppers can choose their components right on the product page.

To add it, open your theme editor, go to the bundle product's template, and add the **TrueBundle picker** block. No code is required, and because it is a theme app block it is removed cleanly if you ever uninstall the app.

Block settings:

- **Heading.** The picker title, for example "Build your bundle".
- **Accent color.** Matches the picker to your brand.
- **Show prices.** Whether component prices appear in the picker.
- **Button label.** The add-to-cart button text, for example "Add bundle to cart".

The picker only renders for a published, active bundle. If the bundle is a draft, or its configuration cannot be read, the picker stays hidden. That is the safe state, and it matches the checkout Function, which also fails closed. A fully fixed bundle needs no picker at all.

Keeping prices in sync

Percent-off pricing and the fixed-total split are computed from the component list-price snapshots stored inside each bundle. If you change a component product's price in Shopify, TrueBundle refreshes the snapshot in every active bundle that uses that variant, so your totals never drift from the prices you actually charge. This happens automatically in the background; there is nothing to run by hand.

What TrueBundle does not do

Stated plainly, so there are no surprises:

- It is **not a subscription-bundle tool**. Shopify's checkout does not allow expanding lines that carry a subscription or selling plan, so TrueBundle is for one-time purchases.
- It is **not an upsell or cross-sell marketing suite**. It is a bundle correctness engine, not a widgets-everywhere upsell app.
- It does **not do quantity breaks, buy-one-get-one, or free-gift promotions**. Those are a different kind of app.

What it does do is the part most bundle apps get wrong: the total always adds up, and a bundle never oversells its parts.

Troubleshooting

A few common situations:

- **The Publish button is not available.** The preflight has a problem to fix. Work through the list in the Preflight checks panel; publishing unlocks when every check passes.
- **The picker does not appear on the product page.** Make sure the bundle is published and active, and that the **TrueBundle picker** block is added to that product's template in the theme editor. Drafts do not render a picker.
- **A percent-off total looks off after a price change.** TrueBundle refreshes the component snapshots automatically, but if you just changed a price, give it a moment. You can also reopen the bundle to confirm the snapshots, then publish again.

- **A list price is rejected in the builder.** Enter the amount with no more decimal places than your store currency uses, for example two for most currencies and none for JPY.

For more, see the FAQ and support page.

Getting help

If something is not working or you have a question this manual did not answer, contact support and include your store URL and a short description of what you expected. We are glad to help.