

The Combined Listings Search Audit

One afternoon. Every combined listing. One honest answer: is this stock findable, or just tidy?

Combined Listings makes your product page cleaner by grouping related products under one parent. It does not decide what your search can find. That is a separate setting, and Shopify's own product filters do not include combined-listing child products in filter results.

This sheet finds the gap. For each combined listing, or each collection that uses them, run five checks and mark pass or fail. Every fail is stock hiding from search behind a tidy catalog.

How to run it

1. List the collections that use Combined Listings. One row per collection.
 2. For each one, run the five checks below against your live storefront, searching and filtering the way a shopper would.
 3. Mark each cell P (pass) or F (fail). Write a quick note where it helps.
 4. Count the fails per collection. Any collection with two or more fails is a discovery job, not a tidiness job.
-

The five checks

- **Child-query.** Search the distinguishing attribute of a child product (its unique color, size, or material). Does the matching item appear? Pick one combined listing and search the term only that child would match.
 - **Facet coverage.** Narrow a collection by a variant facet ("Color: Red"). Do the combined-listing children survive the filter, or do they silently drop out? Shopify's Search & Discovery filters exclude combined-listing children, so check this one carefully.
 - **Parent-only trap.** Check your Search & Discovery setting for combined listings (child-only, parent-only, or both). If it is parent-only, confirm child-specific queries are not being hidden.
 - **AI surface.** Phrase the same child-specific query the way an AI shopping agent would ("a red one in medium") and run it. Does a result come back, or a dead end?
 - **Parent-vs-breakout.** Decide, for this collection, whether the surface should show one parent tile or individual variant tiles. Write down the call.
-

Your per-collection grid

Swap the example collections for your own. Add rows as needed.

Collection	Child-query	Facet coverage	Parent-only trap	AI surface	Parent-vs-breakout	Fails
Sneakers (8 colors)	☐	☐	☐	☐	☐	—
Sofas (6 fabrics)	☐	☐	☐	☐	☐	—
Rings (stone options)	☐	☐	☐	☐	☐	—
Tees (size + color)	☐	☐	☐	☐	☐	—
_____	☐	☐	☐	☐	☐	—
_____	☐	☐	☐	☐	☐	—

Total fail cells across all collections: ____

What each fail means, and where to fix it

- **Child-query fail.** A shopper searching a child's color, size, or material gets nothing. The variant layer is not findable. Make that attribute searchable as a variant-scoped attribute, and break the product out into variant tiles so the exact variant can return as its own result.
 - **Facet coverage fail.** A variant facet drops your combined-listing children. This is Shopify's documented filter behavior, not your mistake. Lean on variant breakout tiles and variant-aware faceting where filtering by a variant attribute is how shoppers shop that collection.
 - **Parent-only trap fail.** Your Search & Discovery policy is set to parent-only and child-specific demand cannot reach the children. Set the policy deliberately per surface: child-only or both where child-specific queries matter.
 - **AI surface fail.** An AI shopping agent's child-specific request hits the same wall a human does. Fixing the child-query and facet failures fixes this one too, because the agent runs the same query against the same surface.
 - **Parent-vs-breakout fail (undecided).** You have not decided how this collection should render. Choose per collection: variant tiles where shoppers browse by color or size, a parent tile for curated or editorial surfaces.
-

Read your score

- **0 fails across all collections:** rare and good. Re-run next quarter and after your next catalog consolidation.
 - **1 to 3 fails:** specific, fixable gaps. Work them collection by collection in the order above.
 - **4 or more fails, or any collection failing 3+ checks:** that collection is tidy but not findable. It is hiding variant-level stock behind a clean parent, and it is likely leaking conversion to a zero-result page.
-

Where the fix lives

This sheet checks what the search layer controls: whether variant-level attributes are searchable, whether the exact variant renders as its own tile, and how your combined-listing policy is set.

Variant breakouts display variants as individual tiles on collection pages and search results. Variant-scoped attributes like options, SKU, and variant metafields become keyword and vector matchable, so variant-specific demand finds your stock.

Your combined-listing parent/child relationship stays where it belongs, in your catalog and Search & Discovery settings. The search layer makes the variant inside it findable.

Want us to run this live on your catalog?

We'll take your combined listings, run the same five checks live against your real catalog, and show you collection by collection what a shopper (and an AI agent) actually finds. 30 minutes, your collections, your products.

[Book a demo →](#)

Built by the Layers team. Enterprise search and merchandising for Shopify Plus. Last updated July 2026.
