

465 SKUs repriced overnight. No receipt total moved.

California retailers price out the door: the shelf number is what you pay, three stacked taxes buried inside it. That convention makes every menu listing look about 39% more expensive than a competitor quoting pre-tax. I converted the whole catalogue in one overnight cutover and found five platform defects on the way through.

I run retail operations at Pacific Wellness in Sherman Oaks, California, and took this cutover end to end on the night of 15 September 2026 — on Blaze POS, with a synced Weedmaps menu downstream of it.

One SKU, both sides	
Before, the shelf read \$65.00 with the taxes inside it.	
Shelf price	46.82
City tax, 10%	4.68
Excise, 15%	7.73
Sales tax, 9.75%	5.77
Customer pays	\$65.00
Identical to the day before, and repeated across 460 repriced SKUs.	

An optics problem with a compliance floor under it

Menu platforms display the shelf price. A store quoting out the door shows a number that already contains all three taxes; a store quoting pre-tax shows a number 38.8% smaller for the identical jar. Side by side on the same screen, we looked expensive because we were the ones being literal.

After the cutover our eighth displays at **\$14.41** where a competitor quoting out the door shows **\$20.00**. Same product, same money at the register.

That last clause is what made this hard. It was not a price change. Every customer had to pay exactly what they paid the day before, to the dollar, because the total is what people remember and what the menu comments notice. The shelf number was allowed to move. The total was not.

The math, and the one flag it all hangs on

The three taxes compound in a fixed order. Each is levied on the running total of the ones before it, and the register rounds the result to the nearest whole dollar.

city = 10% × sticker

excise = 15% × (sticker + city)

sales = 9.75% × (sticker + city + excise)

total = rounded to the nearest whole dollar

combined multiplier = $1.10 \times 1.15 \times 1.0975 = 1.3883375$

Every old price divided by 1.3883375 gives the pre-tax sticker that rings the same total. The entire migration is that one division — provided the platform is actually compounding in that order.

One setting controls it: *calculate excise tax after city tax*. Switched off, the multiplier becomes 1.371875 and every price in the store is wrong by 1.2% — small enough to survive a spot check, large enough to be a year-long tax exposure. I verified that flag first and never touched it again.

Nearest-dollar rounding does one favour: it absorbs the cent of arithmetic drift across the catalogue, so no SKU needed hand-tuning. It also hides a defect, which is defect five.

What changed in the platform

Seven company-level settings, three consumer types across two tax tabs, and a mass update on every inventory record. Four of the seven changed. The other three had to be confirmed unchanged, which is different work and just as easy to get wrong.

Setting	Value	Status
Includes state excise	Enabled	unchanged
Excise line item on receipt	Enabled	changed
Excise taxes included in total	Disabled	changed
State excise rate	15.00%	unchanged
Calculate excise after city tax	Enabled	load-bearing, do not touch
Product prices include excise, default	Disabled	changed
Surcharge subject to excise	Disabled	unchanged
Adult use: city and sales	10% and 9.75%, both post-taxed	sales row taxes excise
Medicinal, both types	City 10% post-taxed, sales 0% inactive	changed
Non-cannabis tab	Sales 9.75% post-taxed, all types	changed
Inventory: price includes excise	No, on all 466 products	mass update

Five defects, and what I did about each

None are documented. All five are reproducible. Two of them will silently corrupt every price in a store, and one hides the evidence on three quarters of the catalogue.

1 The importer rejects its own exporter's output

The product importer rejects non-ASCII characters in the name field, including names the platform's own exporter had just produced. Seventeen products carrying `™`, `é` or `ñ` bounced. Only the name field is validated this way — description passed with non-ASCII intact on 127 rows, brand on one.

What I did. Transliterated names to ASCII in the import file, left description and brand alone, and kept the original strings so display names could be restored afterward.

2 The flag that decides the price lives where the import can't reach it

Whether a price is tax-inclusive is stored on the store inventory record, not the company product. The company product page has no such field and the 38-column import file has no such column. So the import cheerfully loads pre-tax prices into records still flagged tax-inclusive.

Consequence if missed. The register reads \$14.41 as tax-inclusive, backs the taxes out to a \$10.38 base, and charges \$14.41 — roughly **28% under** on every SKU in the store, with no error raised anywhere.

What I did. Ran it as a separate mass update across all 466 inventory records, and verified it before loading a single price.

3 The tax form renders stale defaults for eight seconds

Open a consumer type for editing and every tax order field shows *pre-taxed* for about eight seconds, regardless of what is saved. Save inside that window and you write those defaults back, silently undoing the conversion you just made.

What I did. Waited for the form to populate, compared it against the read-only view, and treated the confirmation dialog as the only trustworthy check — it enumerates exactly what is about to change, which the form itself does not.

4

Saves fail silently

One save was lost to a dropped connection with no error surfaced and a success state rendered anyway. I caught it only because I was reloading between steps.

What I did. Made reload-and-read-back-from-server the rule after every save. A success message and the post-save render are both client-side optimism, and neither is evidence.

5

Rounding conceals a tax misconfiguration on 76% of the catalogue

Because totals round to the nearest dollar, **349 of 460 SKUs ring an identical total** whether or not sales tax is correctly levied on the excise. Only 111 expose the difference. My first test item rang \$35.00 under both configurations — a 41-cent error that vanished into rounding.

What I did. Stopped verifying against totals. Verification reads the sales tax line on the receipt: on a \$46.82 item it must read \$5.77, not \$5.02. A total-based check would have passed a broken store.

Verification

Six SKUs spread across the price range, rung up without completing the transactions, plus one discounted item — discounts now come off the pre-tax base rather than a tax-inclusive price, so they needed their own check.

SKU	Product	Sticker	Must ring
FLO-09a1479b8	Acai Runtz Smalls 3.5g	\$14.41	\$20.01
FLO-0ee75676d	Acai Runtz Smalls 7g	\$25.21	\$35.00
PF7APRO1	Autumn Brands Preroll 14pk	\$44.66	\$62.00
P8B6PRO1	Birdies Classic Hybrid 3pk	\$10.08	\$13.99
EDI-700ef113f	Breez Extra Strength Nighttime	\$46.82	\$65.00
VAP-415ef2e68	Cizi 1G 510 Cart Blue Dream	\$12.97	\$18.01

Two known failure signatures. A total about 28% low means the excise-inclusive flag is still set on inventory. Totals off by about 1.2% mean the compound-ordering flag got switched off.

Promotions were the second migration

Promotions apply to the pre-tax subtotal. Every dollar figure in every active promotion had been written as an out-the-door figure, so each one silently became 38.8% more generous the moment the catalogue converted.

The rule that fell out of it: **if a promotion names a dollar amount, it is broken; if it names a percentage or a ratio, it is fine.** Percentage discounts, BOGO, buy-two-get-one and penny items all scale themselves. Basket minimums, fixed dollar discounts, fixed dollar deal prices, loyalty tier breaks and flat-priced bundles all had to be divided through.

Promotion	Was	Now
Loyalty basket minimum and discount	40.00 / 5.00	28.80 / 3.60
Lighters, two for	5.00	3.60
House flower 7g	35.00	25.21
House flower 14g	65.00	46.82
House flower 28g	120.00	86.43
Big Nugs 7g	55.00	39.62
Big Nugs 14g, three configurations	100.00	72.03
Big Nugs 28g, five configurations	180.00	129.65
Joints, five for	15.00	10.80

Why the basket minimum is \$28.80 and not \$28.81

\$40.00 divided by 1.3883375 is \$28.8115. Rounding down rather than up costs nothing and buys a cent of slack, which matters because our best-selling all-in-one vape lands at exactly \$28.81 pre-tax and would otherwise sit precisely on the threshold. The platform does not document whether the comparison is `>` or `>=`, so I designed so that it cannot matter.

The discount side is the same reasoning in reverse. A \$3.60 pre-tax discount reaches the register as \$5.00 off and holds at every basket size. A \$5.00 pre-tax discount would have handed the customer \$6.94 of value and read as \$7.00 off — a 39% overrun on a promotion nobody would have thought to re-check.

Rollback, and closing the tail

The cutover ran with a written rollback in hand: re-import the preserved price file, re-enable the excise default, return every tax order to pre-taxed across both tabs and all three consumer types, set the adult-use sales row back to not taxing excise, restore the receipt line item and the included-in-total flag, then mass update inventory back to tax-inclusive — reloading and verifying after each step. It was never needed. It existed so that the decision to proceed at eleven at night was a cheap one.

The three items left open at dawn, and how each closed

- **The 466th product.** There were 466 active inventory products and only 465 in the export the import was built from, so one was ringing about 39% high somewhere in the catalogue. Finding it took one sort: every converted price lands on an odd cent, so any whole-dollar figure that isn't a known exception is a survivor of the old basis. It turned out not to be a timing gap at all — it was a 1.5g preroll I had created twice. The discrepancy that looked like a race against the export was a duplicate record wearing a race's clothes, which is worth knowing the next time a count comes up one short. Corrected.
- **The sample SKU** parked at a test price is back to a penny.
- **Customer-facing copy has caught up.** Menu banners, the store's about section, the deal that had been titled around a tax-inclusive price, the first-time offer, the second menu platform, the webstores, the business listing and in-store signage all now quote the same basis the register does.

The migration is closed. What follows is the part that outlives it.

What I'd change in the product

I ran this as an operator because there was nobody else to run it. But four of the five defects are product decisions rather than bugs, and none of them would be expensive to reverse.

- Tax-inclusive versus tax-exclusive should be a first-class migration primitive with a dry run and a diff, not a checkbox in company settings that several unrelated record types quietly depend on.
- Any field that can put a whole catalogue 28% under should be reachable by the same mechanism that sets the prices. The excise-inclusive flag belongs in the product import schema.
- A form must not render plausible, wrong, editable values before server state resolves, and a save confirmation should be a server read-back rather than optimistic UI. Eight seconds of a spinner would have been better than eight seconds of a lie, and if the platform can't confirm the write it shouldn't draw the success state.
- Verification tooling should surface per-tax-line deltas rather than totals. Totals hid this defect on 76% of the catalogue; a tool that compared tax lines would have caught it on the first item, which is the difference between an hour of work and a week of wrong prices nobody noticed.

The general case is the promotions one. A platform that lets a price mean two different things, and then stores only the number, has handed every migration risk to whoever happens to be standing at the counter. Storing the basis alongside the amount is a one-

column change that would have converted nine promotions automatically instead of leaving an operator to find them by hand at midnight.

Kyle Williams

Managing Partner, Pacific Wellness — Sherman Oaks, California. Retail operations, purchasing, financials, POS administration and analytics. Fifteen years in cannabis, from delivery in 2011 through a decade of wholesale brokering.

[linkedin.com/in/kylewilliams021](https://www.linkedin.com/in/kylewilliams021)