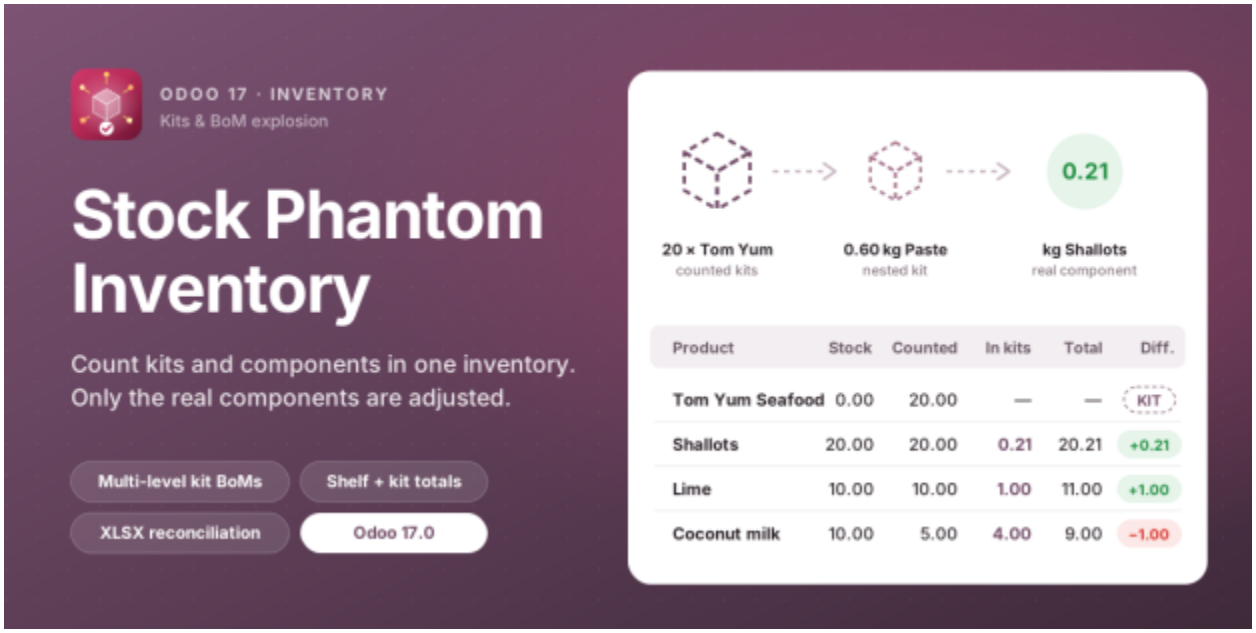


Smart Kit Inventory Adjustment (Step-by-Step Case Study)

Part 1

This guide demonstrates how the **Smart Kit Inventory** module automatically handles nested **Kit Bills of Materials (Phantom BoMs)** during a physical stock take.

In this real-world hospitality scenario, we will look at how counting finished dishes on the kitchen floor automatically adjusts the balances of raw ingredients hidden inside them.

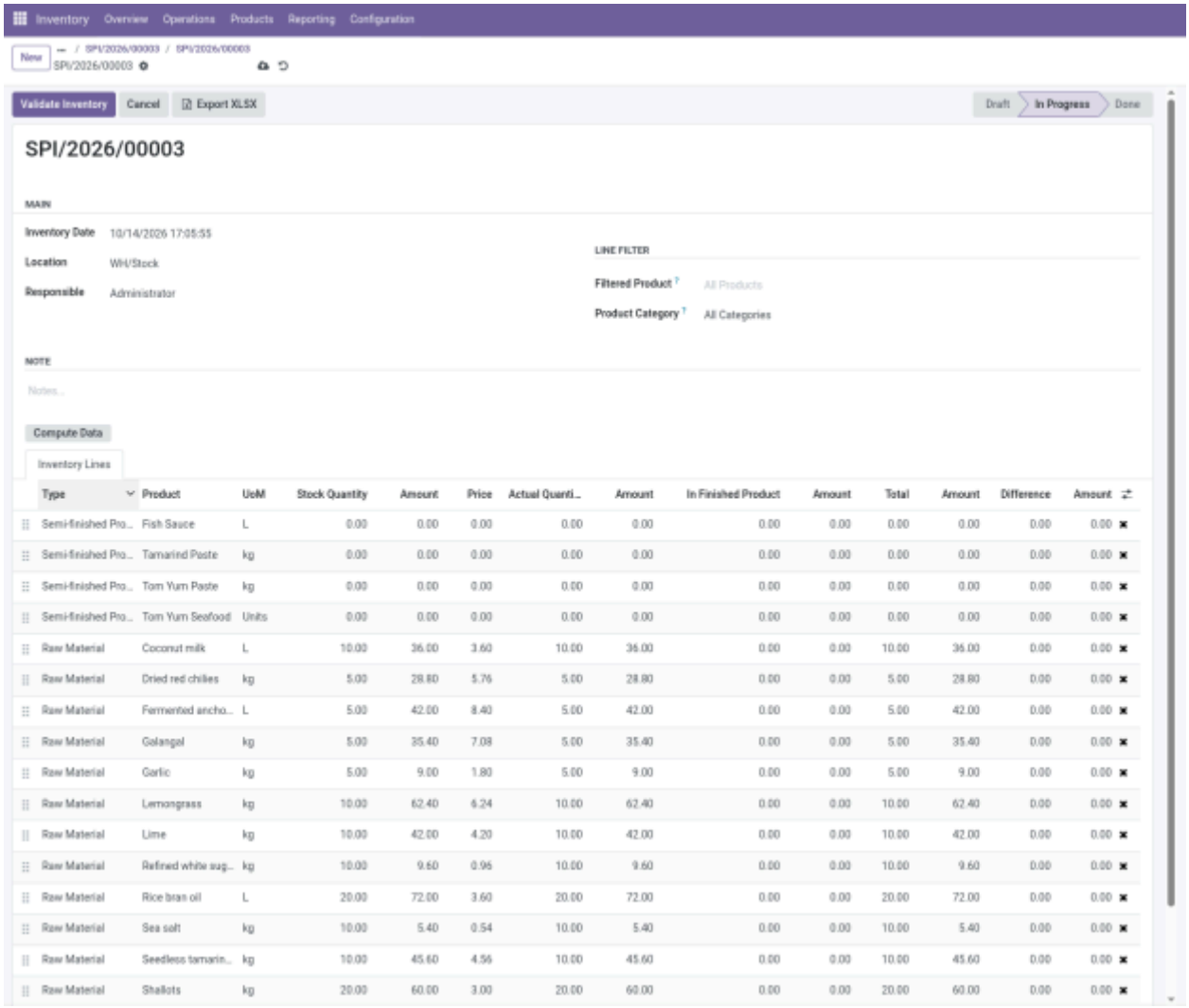
Step 1: Baseline Stock State (Before Inventory)

Before starting the inventory audit, our storage location (**WH/Stock**) holds raw materials, but has **zero** pre-made kits recorded in system book stock.

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

Here is our initial warehouse snapshot:

- **Raw Materials:**
  - `Coconut milk`: **10.00 L**
  - `Tiger prawns`: **20.00 kg**
  - `Shallots`: **20.00 kg**
  - `Rice bran oil`: **20.00 L**
- **Prepared Kits & Subassemblies:**
  - `Tom Yum Seafood` (Finished Dish): **0.00 Units**
  - `Tom Yum Paste` (Pre-made Subassembly): **0.00 kg**

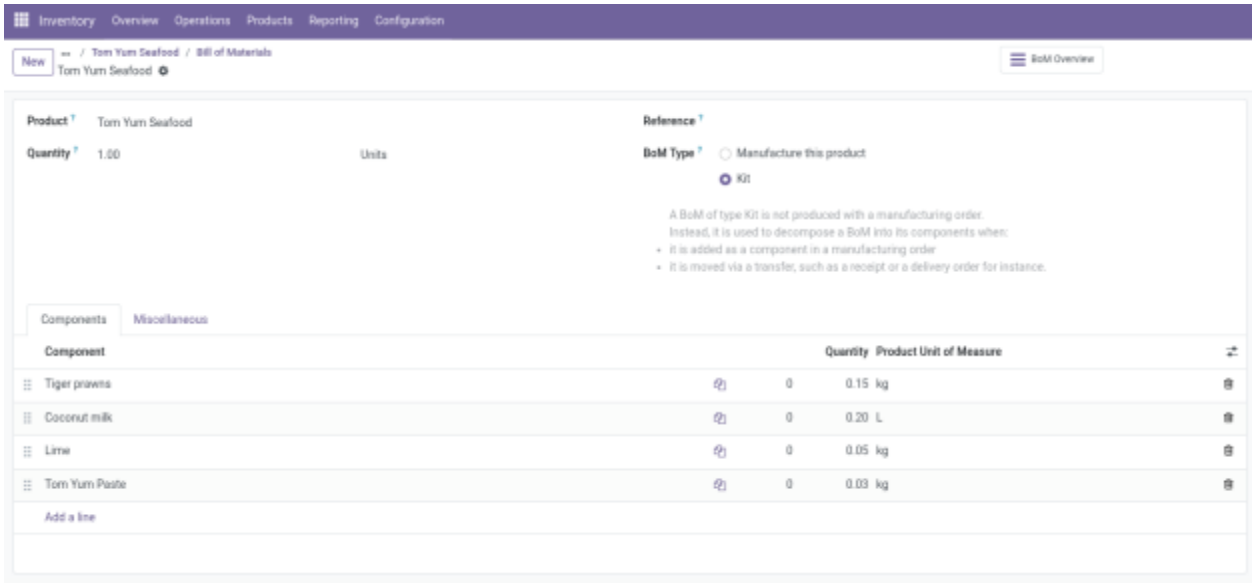


This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

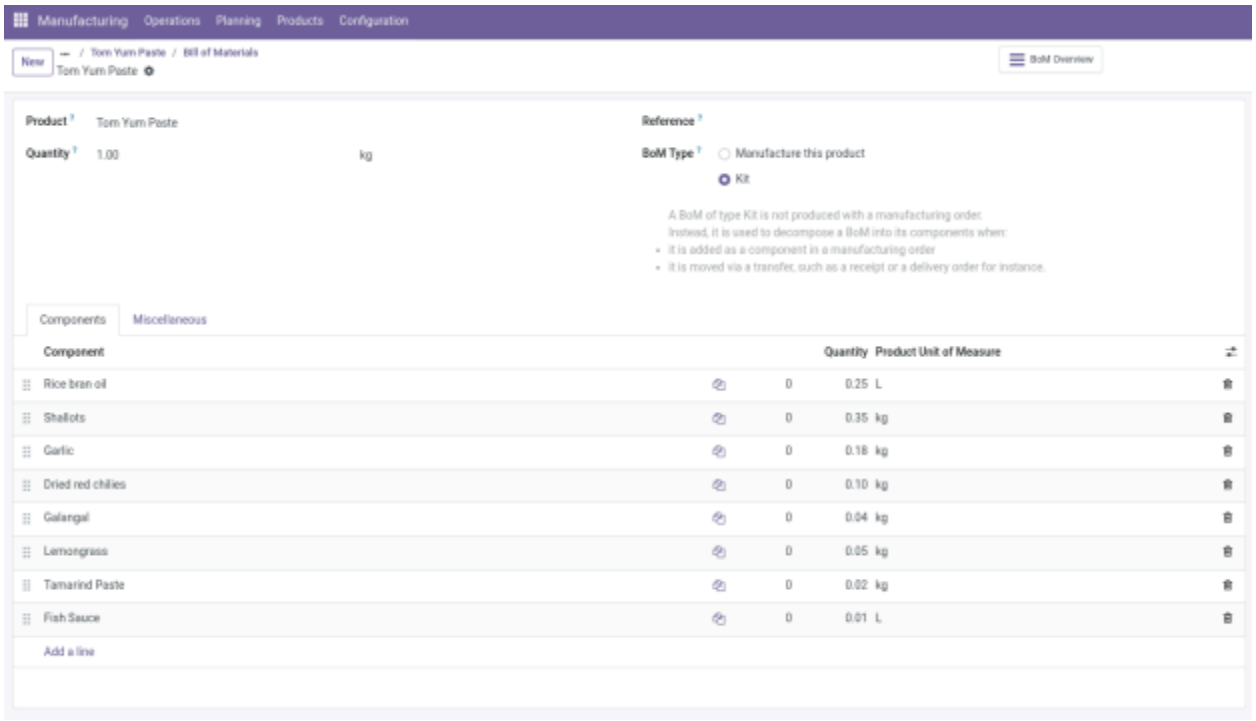
The Nested BoM Structure

Our dishes use multi-level, nested Kit specifications:

1. **`Tom Yum Seafood` (1 Unit)** requires `0.03 kg` of **`Tom Yum Paste`** as a component.
2. **`Tom Yum Paste` (1 kg)** requires `0.35 kg` of raw **`Shallots`** as an ingredient.



This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.



This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

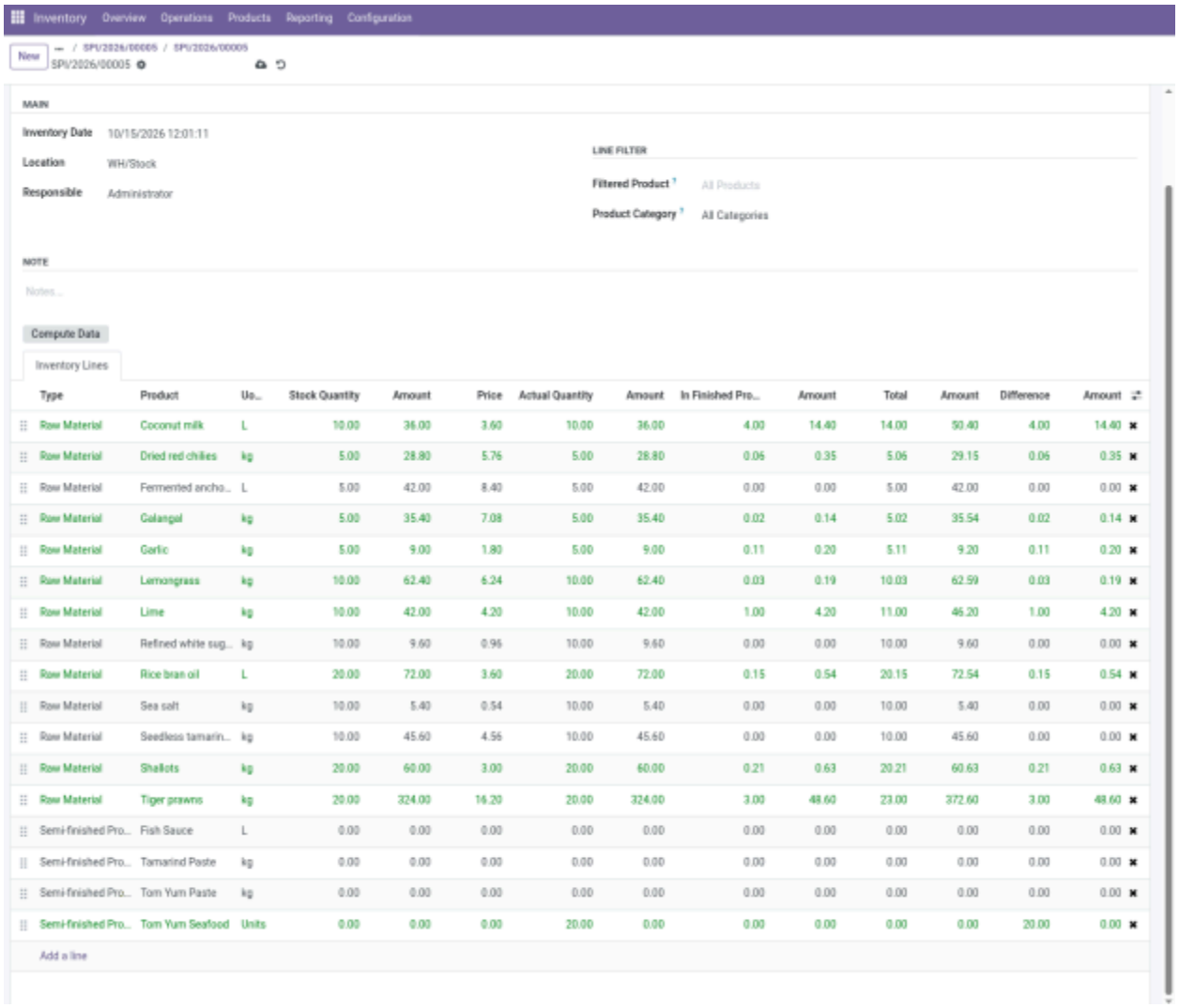
Step 2: The Physical Count & Smart Recomputation

During a physical stock take at the end of the shift, the kitchen supervisor finds that **all raw ingredient packages on shelves are completely intact**, but the chefs have already pre-assembled **20 finished units of Tom Yum Seafood** sitting on the counter.

What the User Inputs:

1. The user clicks **Compute Data** to fetch system baselines.
2. For raw ingredients (like `Shallots`), the user enters the exact amount seen on shelves: **`20.00 kg`**.
3. For the finished dish, the user enters the physical count: **`20.00 Units`** of `Tom Yum Seafood`.

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.



This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

Deep Dive: The Mathematical Calculation for "Shallots"

Let's trace how the engine calculates the metrics for **Shallots** behind the scenes to display them on the data grid:

1. Recursive BoM Explosion

Since `Tom Yum Seafood` contains a nested kit (`Tom Yum Paste`), the module drills through the hierarchy to evaluate the raw ingredients:

$$Total\ Paste\ in\ 20\ Soups = 20.00\ Units \times 0.03\ kg\ (Paste\ per\ Unit) = 0.60\ kg\ of\ Tom\ Yum\ Paste$$

$$Shallots\ hidden\ inside\ that\ Paste = 0.60\ kg\ (Paste) \times 0.35\ kg\ (Shallots\ per\ kg) = 0.21\ kg$$

The module automatically updates the **In Finished Product** column for Shallots to `0.21`.

2. Total Quantity Calculation

The system combines loose raw stock found on shelves with the computed volume hidden inside the prepared kits:

$$Total\ Quantity = Actual\ Quantity\ (Shelves) + In\ Finished\ Product\ (Kits)$$

$$Total\ Quantity = 20.00\ kg + 0.21\ kg = 20.21\ kg$$

3. Final Variance (Difference) Calculation

Finally, the module compares the compiled total against the starting system book stock to calculate the real variance:

$$Difference = Total\ Quantity - Stock\ Quantity\ (System\ Book)$$

$$Difference = 20.21\ kg - 20.00\ kg = +0.21\ kg$$

UI Visual Feedback

Because a surplus variance of `+0.21 kg` (valued at `$0.63`) is detected, the **Shallots row highlights in Green** in the user interface, signaling a positive stock adjustment.

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

Step 3: Validating the Inventory Adjustment

When you click **Validate Inventory**, the module applies Odoo stock corrections with strict functional guardrails:

- **Raw Components:** Odoo posts a physical correction of `+0.21 kg` to the `Shallots` stock ledger, aligning system quantities with real-life volume (20.21 kg total).
- **Kits/Subassemblies:** The system stock for `Tom Yum Seafood` **remains at 0.00 Units**. Because it is a Kit, it never holds direct inventory; its value is entirely stored within its raw components.

Your inventory valuation layers (`stock.valuation.layer`) remain balanced without manual kitchen math!

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

Part 2

The Physical Count & The "Coconut Milk" Case Study

During the physical stock take, the warehouse manager counts what is physically left on the shelves and what has been pre-produced by the kitchen staff.

What the user enters on the screen:

1. `Tom Yum Seafood` **(Kit):** `20.00 Units` (The manager finds 20 ready-to-serve portions in the kitchen).
2. `Coconut milk` **(Raw Material):** `5.00 L` (The manager counts only 5 unopened liters left on the ingredient shelves).

🔍 Behind the Scenes: How Odoo Calculates the Rows

Let's look at the row for **Coconut milk** on the screenshot and break down the math performed by the module:

1. Automated BoM Explosion (`In Finished Product`)

The module sees that you have 20 units of `Tom Yum Seafood`. It opens its Bill of Materials, finds that 1 unit requires `0.200 L` of coconut milk, and calculates:

$$In\ Finished\ Product = 20.00\ Units \times 0.200\ L = 4.00\ L$$

(This tells the system: "4 liters of milk are not lost; they are physically inside those 20 cooked soups").

2. Recomputing the Absolute Total (`Total`)

The module combines the loose stock on the shelves with the stock hidden inside the kits:

$$Total = Actual\ Quantity\ (Shelves) + In\ Finished\ Product\ (Kits)$$

$$Total = 5.00\ L + 4.00\ L = 9.00\ L$$

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

3. Finding the Real Operational Discrepancy (`Difference`)

Standard Odoo would compare your shelf count (5.00 L) directly against the book stock (10.00 L) and claim you lost 5 liters. **Our module acts smarter.** It compares the *true total* against the system book stock:

$$Difference = Total - Stock\ Quantity\ (Book\ Stock)$$

$$Difference = 9.00\ L - 10.00\ L = \ -1.00\ L$$

🔴 The UI Visual Outcome (Why this makes sense):

- **The row highlights in red:** The system detects a real shortage of exactly `-1.00 L` (valued at `-$3.60`).
- **The business logic:** This means the kitchen didn't just use 4 liters for the soups; they actually wasted or spilled 1 extra liter of coconut milk during the shift. The module successfully isolated the real theft/waste from the legitimate production usage!

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.

Important Conclusion: What Happens Upon Validation?

When the warehouse manager clicks the **Validate Inventory** button, the module processes the audit sheet using strict Odoo functional guardrails:

- **NO "Tom Yum Seafood" units will be created or added to your warehouse stock.** Since `Tom Yum Seafood` is configured as a **Kit**, it cannot physically hold inventory or quants in Odoo. Its system stock will remain unchanged.
- **The system dynamically adjusts ONLY the raw ingredients.** Instead of forcing you to create complex Manufacturing Orders, the module directly updates the physical ledger of the underlying components based on the **Total** column.

Post-Validation Ledger Impact (Our Case Study):

- **Coconut milk:** Odoo automatically posts an adjustment move for `-1.00 L` (setting the final on-hand stock to `9.00 L`).
- **Tiger prawns:** Odoo registers `3.00` variance. System stock remains `20.00 kg` (17 kg on the shelves + 3 kg inside the soups).
- **Shallots:** Odoo records a positive variance of `+0.21 kg`, correcting the total available stock to `20.21 kg`.

The Bottom Line: Your ingredient inventory levels and financial valuation layers (`stock.valuation.layer`) are corrected in one single click, without cluttering your database with fake finished goods entries!

Support

e-mail: support@guscom.by

website: https://www.guscom.by

This document is the property of JLLC Guscom software and consulting and may not be modified without its consent.