SHRINK SOFTWARE

ORIGIN STORY

Why Shrink Software Exists.

An origin story written by the team that lived the problem before they built the product.

PUBLISHED BY

Shrink Software LLC

AUDIENCE

Operators, partners, and investors who want to understand the operational experience behind the product

DATE

May 2026

FORMAT

Narrative — a 15-minute read explaining why we built what we built

ORIGIN STORY

This product was built by the people who had to use it.

Most enterprise software for institutional food production is built by software companies who studied the regulations and interviewed operators. Shrink Manager was built differently. The team that built it ran a real institutional food operation for years before deciding the available software was structurally incapable of doing the job. The product exists because we needed it ourselves and could not buy it.

This document explains how that happened, what we learned from running an operation while building software for it, and what the experience produced that we believe other operators will recognize.

It is written in collective voice — Shrink Software team, Anu Sushi team — rather than in individual founder voice, because the work was collective. Multiple operators, engineers, food safety professionals, and operations leaders contributed to the realizations described here. The decisions made and the product built reflect that team's lived experience, not any single founder's vision.

THE OPERATIONAL REALITY

Anu Sushi LLC: where the problem became clear.

Anu Sushi LLC is an institutional wholesale sushi manufacturer. The company supplies fresh sushi production through institutional foodservice channels — health systems, university dining, corporate campuses, and managed services accounts. As of 2026, the operation runs across multiple production sites, producing more than one million units per year. Compliance output reaches over 200 institutional buyer accounts through partner distribution channels including Aramark, Sodexo, and Compass Group.

For most of the company's history, this was hard but tractable work. The food safety requirements were known. The operations could be run. The financial controls existed in a standard ERP. The system did not pretend to capture every workflow detail, and nobody expected it to.

Then three things happened in sequence.

1. The operation got bigger

Multi-site operation introduced a kind of complexity that single-site operation does not. When a problem appeared at one site — a shrink anomaly, a buyer complaint, a missing record — the answer to "what actually happened" required reconstructing operational reality from paper logs, partial emails, and the recollections of shift leads. Reconstruction worked when there was time. There was rarely time.

2. The buyers got more demanding

Health systems, university dining programs, and managed services operators began requiring traceability documentation for fresh food deliveries. The requests started as supplier qualification questionnaires. They evolved into supplier audit programs. By 2023 they had become standing requirements: produce the records or lose the contract. The records were difficult to produce because they didn't exist as records. They existed as work that had happened, in paper logs that were not reconciled, in shift leads' memories.

3. FSMA 204 became real

The FDA's Food Traceability Final Rule, originally with a January 2026 compliance deadline, made buyer pressure into regulatory pressure. The same records the buyers wanted, the FDA would also want — and the FDA would want them in 24 hours. Anu Sushi's team realized that the systems and processes that had worked for years were not going to survive the regulatory environment that was coming.

> **The operational realization.**
>
> The team did not start out trying to build software. The team started out trying to solve their own buyer audit problem and their own coming FDA problem. The software emerged when the search for an existing solution failed.

THE WRONG ANSWER WE TRIED FIRST

The $80,000 NetSuite implementation that didn't work.

Like most institutional food operators facing this problem, Anu Sushi's first instinct was to extend the ERP. NetSuite was already running the financials. NetSuite was a respected platform with food industry references. NetSuite had modules that, according to the marketing, would handle production tracking, lot management, and quality control. The path of least resistance was to use what we already had.

An $80,000 implementation was scoped, approved, and executed. The implementation team was competent. The NetSuite platform was capable of doing what NetSuite was designed to do. The result was an expensive, customized ERP installation that produced exactly the records NetSuite was architecturally capable of producing — and could not capture what the floor actually needed to record.

The structural problem was the one we now describe at length in our published architecture brief, *Why ERPs Fail at the Production Floor*. NetSuite is a transactional system optimized for office-level work. The institutional sushi production floor is not an office. It is a physical environment with operators wearing gloves, ingredients moving through workstations, cooling cycles running on the clock, and labels being applied at the moment of finished-product completion. The transaction model and the workflow reality did not align.

The implementation produced compliance-looking records that came from end-of-shift data entry, not from the work itself. Under realistic audit scrutiny, the timestamps on the records would not match the work they purported to describe. Under buyer audit scrutiny, the records would not survive a question that probed real workflow detail. The implementation was technically successful and operationally inadequate.

WHAT WE LEARNED FROM THE FAILURE

The records have to be the work, not a description of the work.

This is the operational principle that emerged from the NetSuite failure. Compliance records that are generated as a separate task — even when the task is automated — will eventually fail under audit because the records and the work do not have the same provenance. Records that are produced as a byproduct of the work itself have unbreakable provenance because the record and the work are the same thing.

BUILDING WHAT WE NEEDED

Why we started building software inside an operation.

After the NetSuite implementation revealed that the path of least resistance was a dead end, the Anu Sushi team faced a decision. The buyer audit pressure was not going away. The FDA deadline was approaching. The systems on the market either had the same architectural problem NetSuite had, or were thin compliance documentation tools that would not capture the underlying operational work.

The decision the team made — to build the system themselves, inside an operating institutional food company — was not because the team thought of itself as a software company. It was because no available software solved the actual problem and the regulatory deadline was real.

Two things made the in-house build viable that would not have been viable for most operators.

The team had software experience

The Anu Sushi leadership team included Daniel Bradish, who had two decades of experience leading multi-tenant SaaS architecture, with substantial Asia operating experience. Daniel had built enterprise software before he joined the food operation. The combination of operational reality and software engineering capability in the same team made the build possible.

Dr. Bradley Oostindie joined as system architect with a PhD-level systems-thinking background. He had architected the human workflow processes that distinguish institutional food production from restaurant production, and the platform integration layer that connects production floor work to financial and accounting systems. The platform's architectural distinctness traces directly to his work.

The team had the operation to test against

Software built without a real operation to test against produces theoretical software. The Anu Sushi production floor served as the live test environment for every workflow decision, every interface choice, every data model question. The receiving log interface was tested by actual receiving staff. The transformation tracking was validated against actual production runs. The cooling logs were used in actual cooling cycles. Every workflow detail that exists in the product exists because it was validated by operators who had no patience for software that did not work.

The combination — software engineering capability inside an operating food business — produced what neither pure software companies nor pure operators could have produced alone. The Shrink Manager that exists today is the result of more than two years of operational use and refinement inside Anu Sushi before it was offered to other operators.

THE PRINCIPLES THAT EMERGED

What the build process taught us.

The product reflects principles the team did not start with. They emerged from the work. They are the principles we now build against and that distinguish how Shrink Manager handles things from how comparable systems handle them.

The operator on the floor is the user, not the manager in the office

Every workflow capability in Shrink Manager has to be usable by floor staff during production. If a feature requires a manager to come from the office and operate it, the feature is wrong. The implication is that the interface has to work on mobile devices, in the language the operator speaks, with one-handed operation while the other hand is doing actual work. This is harder than it sounds and most software systems fail at it.

Compliance is the byproduct, not the goal

If the goal of the system is compliance, the system will produce compliance records that pass audits and miss the operational value. If the goal of the system is to make the operation run better, the system will produce compliance records as a side effect and also produce shrink visibility, training intelligence, and operational learning that compound over years. We have written about this distinction in our operational brief, *The Hidden Cost of Shrink in Multi-Site Food Operations*.

The ERP stays

The NetSuite failure did not teach us that ERPs are bad. It taught us that ERPs are good at what they are designed for, and bad at what they are not designed for. The right structure is to keep the ERP and add the execution layer alongside it. The integration boundary between the two systems is well-defined and bounded. Operators do not have to choose between their existing financial system and FSMA 204 capability — they can have both.

The product has to extend beyond food

The structural problem Shrink Manager solves — workflow-gated execution capture in regulated production environments — is not specific to food. The same pattern serves pharmaceutical manufacturing, medical device production, regulated packaging, and other production verticals where regulators audit work and missed steps cost recalls, fines, or lives. We built the architecture to extend across these verticals from the start. The patent portfolio reflects this design decision.

WHERE WE ARE TODAY

The state of the company in 2026.

As of the date of this document, Shrink Manager has been in continuous production for more than two years. The platform runs across 10 institutional fresh food production sites operating under Anu Sushi LLC. It processes more than one million units of production per year. The compliance output flowing through Shrink Manager reaches over 200 institutional buyer accounts via partner distribution channels including Aramark, Sodexo, and Compass Group.

The operational results inside Anu Sushi tell us the system works. Shrink is down 15% across the production network. Compliance-related labor is down 30%. Revenue is growing 35% year-over-year, partly because the buyer audit capability the platform provides has opened accounts that were not accessible before.

The company has nine USPTO patent-pending innovations covering workflow-gated execution, temporal authenticity enforcement, and dual-source corroborative verification. The patent portfolio was filed broadly because the architecture serves operations beyond food — pharmaceutical compounding, contract manufacturing, regulated packaging, aerospace, and other audit-exposed production environments share the same structural pattern.

The company is now offering Shrink Manager to external operators. The first external customer engagements are scheduled to close in Q3 2026. The expansion is happening in parallel with continued internal use at Anu Sushi, which means the platform stays grounded in operational reality even as it serves more operators.

WHY THIS STORY MATTERS

What an operator-built product means in practice.

The "built by operators" claim is common in food technology marketing. It is often false. Many software products that claim operator origins were actually built by software professionals who interviewed operators. The interview-based development model produces software that approximates operator workflow without capturing it.

The difference between interview-based and use-based development shows up in workflow details that only matter to people who have done the work. A few specific examples from Shrink Manager:

- **Receiving with cold ingredients.** The receiving interface accommodates the reality that fresh fish arrives partially frozen, that lot codes are sometimes on the inside of the packaging, and that the operator's gloves are wet. None of this matters until you have to receive fish, at which point all of it matters.
- **Production order changes mid-run.** The transformation logic accommodates the operational reality that a production order sometimes changes mid-run when a customer calls in additional units. The system can adjust the input lot allocation without breaking the traceability chain. Most systems force the operator to choose between accurate traceability and operational flexibility; we built the workflow to do both.
- **Cooling cycle interruptions.** Cooling cycles get interrupted in real operations — power events, equipment alerts, operator decisions to extend or shorten the cycle based on visual inspection. The cooling log accommodates these interruptions while preserving the integrity of the underlying record. The interface lets operators record why a cycle was modified, not just that it was.

- **Multilingual operation on the floor.** A meaningful fraction of institutional food production floor staff is more comfortable in Spanish or other languages than in English. Shrink Manager works in the operator's preferred language at the workflow level, not just at the menu level. The data captured is the same regardless of interface language.

None of these details would have been built into a product designed by interviewing operators. They were built into Shrink Manager because the team building the product was simultaneously the team running the operation, and the product had to actually work in production every shift, every day.

THE OPERATOR-BUILT CLAIM IN ONE SENTENCE

You can tell which products were built by operators because the products handle the edge cases operators actually encounter.

The interview-based products handle the canonical workflow that operators describe when asked. The use-based products handle the messy reality operators do not think to describe because they have learned to work around it. Shrink Manager is the second kind because the team building it was working around the same messy reality.

WHAT'S NEXT

Where we go from here.

The company is in a transition. For the first two-plus years of its operating life, Shrink Manager was an internal system for Anu Sushi. For the next several years, the work is to deploy it across other institutional food operations, then across adjacent regulated verticals.

The food production work is the wedge. Institutional fresh food production is where we have the most operational experience, the most product-market fit evidence, and the most regulatory urgency. We expect to add named external food production customers through 2026 and 2027, scaling the deployed footprint deliberately.

The adjacent vertical work follows. Pharmaceutical compounding pharmacies, medical device contract manufacturers, regulated packaging operations, and other audit-exposed production environments share the same architectural pattern. The patent portfolio supports this expansion. The work to bring Shrink Manager into these adjacencies will happen in partnership with operators in those industries — the same use-based development model that built the food product will build the pharmaceutical product, the medical device product, and the others.

The international work is in scoping. Multiple international markets — Mexico, the Netherlands, the United Kingdom, Singapore — have institutional food operators facing similar regulatory environments. International expansion will happen through channel partnerships with local operators who understand their markets better than we do.

None of this is rushed. The principle that guided the build inside Anu Sushi — that the records have to be the work, that the operator on the floor is the user, that the system has to actually work in production every day — applies to every expansion. We expect to add operators at a rate that lets us maintain product quality, not at the rate that maximizes near-term revenue.

SECTION 7

If this story sounds familiar.

If your operation has tried to extend an ERP to handle production floor work and discovered the limits of that approach, the experience described in this document will be familiar. If your team is currently trying to make spreadsheets and paper logs survive an FDA traceability audit, this experience will be familiar. If you are operating across multiple sites and cannot answer the simple question of where your shrink is coming from, this experience will be familiar.

The team at Shrink Software has lived these specific problems. The product was built to solve them because we needed it solved for our own operation. We now offer it to other operators because the architecture we built generalizes — the same workflow-gated execution layer that solved Anu Sushi's compliance and shrink problems will solve the same problems for institutional food operators with similar profiles.

If you'd like to explore whether Shrink Manager fits your operation, the easiest first step is a 30-minute conversation in which we walk through your specific operational situation. We don't run sales-style demos because the product is meant to solve specific problems, and the conversation is more useful if we focus on yours.

Shrink Software LLC
hello@shrinksoftware.com · shrinksoftware.com

This origin story is published by Shrink Software LLC. Operating metrics cited (1M+ units per year, 10 production sites, 200+ institutional accounts, 15% shrink reduction, 30% compliance labor reduction, 35% YoY revenue growth) reflect Anu Sushi LLC operations as of May

2026. Anu Sushi LLC is a separate legal entity that operates as the incubating operator for Shrink Manager. The relationship between the two companies is disclosed in Shrink Software's investor materials.