CASE STUDY

Fixing RCM for a Genetic Lab That Outgrew Its Own Processes

RCM Integration · Prior Auth Automation · Denial Management

Executive Summary

Top 5 Problems

1. Manual double data entry between Salesforce and seeQer — no system integration
2. Prior auth submissions inconsistent across reps — no standardization
3. Claims submitted blind — no pre-submission validation
4. 835 denial files processed manually — appeals missed or delayed
5. No revenue visibility across systems — leadership flying blind

5 Impacts Those Problems Created

1. Transcription errors and time delays killed claims before they started
2. Payor kickbacks multiplied as the team scaled — each kickback added days to the cycle
3. Denial rate compounded fast — 10% at 500 claims is manageable, at 2,000 it's a crisis
4. Recoverable revenue became write-offs when appeal windows were missed
5. No cash flow predictability — every operational decision made with incomplete data

Solution We Built:

Bidirectional Salesforce ↔ seeQer integration + Pre-submission CPT-ICD validation + Automated 835 denial parsing and appeals routing

Impact of Our Solution

- Revenue turnaround: beyond 3–4 weeks → 1–2 weeks
- Prior auth: manual and inconsistent → automated and standardized
- Denial rate: materially reduced via upstream validation
- Rep workflow: two systems, manual toggle → single interface, real-time sync
- 835 intake: manual portal checks → automated parsing and queue routing
- Revenue visibility: fragmented → real-time single source of truth
- Scaling model: headcount → system capacity

A Lab That Grew Faster Than Its Processes

Growth is supposed to be the goal. For this genetic testing laboratory in North America, it arrived fast, headcount doubled from 50 to over 100 people in a matter of months. New provider accounts came on board. Test volumes climbed. Revenue should have followed.

It did not. Not cleanly.

The processes that held the lab together with 50 people were never designed to scale. Insurance case tracking, prior authorization submissions, claims management, all of it ran on institutional knowledge and a team small enough that nothing fell through the cracks. When the team doubled, that institutional knowledge became a fraction of what it needed to be. New hires inherited the same workflows without the same context. And the cracks started showing.

Claims that used to be resolved in days sat for weeks. Prior auth submissions went out inconsistently. Denial rates crept up. Revenue turnaround stretched well past the standard three-week window. Customer care reps spent more time toggling between systems and re-entering data than actually helping patients.

The lab tried the obvious fix. They hired more people. It did not work.

> **The processes that worked at 50 people started breaking at 100. New hires inherited the same workflows without the same context. And the cracks started showing.**

What Breaking Actually Looked Like

Institutional Knowledge That Did Not Scale

At 50 people, three reps handling prior auth develop informal consistency. They know which payors need what, which documentation gets flagged, and which edge cases to watch for. At 100 people with higher volumes, that consistency disappears. Different reps document cases differently. Payors kick submissions back for gaps that never appeared when fewer people were handling the same work. Each kickback adds days to the revenue cycle.

Double Data Entry as an Error Factory

Every time a rep copied patient demographics, insurance details, and clinical indications from Salesforce into seeQer manually, two risks emerged: transcription error and time delay. In RCM, both kill revenue. A wrong ICD code means a denial. A delayed prior auth means a held claim. The reps were not incompetent; they were doing what the system forced them to do.

Finding Out a Claim Will Be Denied After Submission

The traditional RCM loop, submit, wait, get denied, appeal, carries real cost. Denial management staff time, appeals processing time, delayed cash collection, and, in some cases, write-offs when appeals are not filed within the window. For a lab scaling fast, this loop compounds quickly. A 10% denial rate at 500 claims a month is a different problem than a 10% denial rate at 2,000 claims a month.

835 Files Sitting Unread

When a payor denies a claim, they generate an 835 Electronic Remittance Advice file explaining why. At scale, this queue grows faster than a team can work through it manually. Appeals that should have been filed within 30 days slip past the window. Revenue that was recoverable becomes a write-off.

No Revenue Visibility, No Operational Confidence

Leadership had no reliable way to forecast cash flow. Claims sat in various states across multiple systems. Prior authorizations were tracked inconsistently. Denial status required manual portal checks. Running a lab, ordering reagents, planning headcount, and committing to provider contracts, without revenue predictability, means every decision is made with incomplete information.

Headcount as the Wrong Answer

More reps for insurance queries, more staff for prior auth and denial follow-up. It did not work. More people doing a broken process produce more broken outputs at a higher cost. Each new hire needed training, inherited the same fragmented systems, and hit the same walls. Labour costs went up. Throughput did not scale proportionally.

What We Found

We spent the first days not writing code. We talked to customer care reps, lab technicians, billing staff, and leadership. The problem looked different depending on where you sat.

Customer care saw double data entry and delayed query resolution. The lab team saw billing-related calls eating into clinical work. Leadership saw unpredictable revenue and no way to plan around it.

We mapped the full patient journey from test ordered through insurance verification, prior auth, claims submission, and payment collection. Every handoff, every system switch, every point where data was re-entered, or status was checked manually. The bottlenecks became clear.

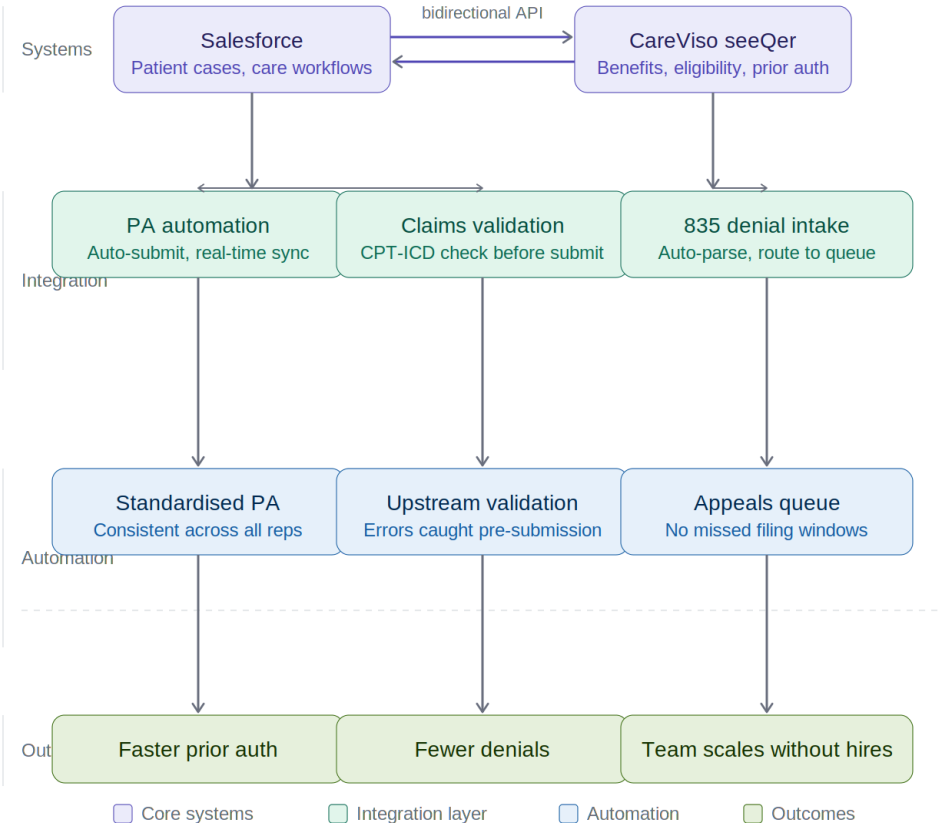
What We Built

1. Salesforce and seeQer Integration, End-to-End

The most immediate pain was the two systems that did not talk to each other. Salesforce held patient cases and customer care workflows. CareViso's seeQer platform handled benefits verification, eligibility checks, prior authorization determination, and PA submission across payors. Between them: nothing. Just people copying data from one screen to another.

We built a bidirectional integration using seeQer's API layer. When a customer care rep creates a case in Salesforce and flags it for prior authorization, the system automatically triggers a seeQer API call, pushing patient demographics, insurance details, test type, and clinical indications into seeQer. It runs eligibility verification, determines whether prior auth is required, and if it is, submits the PA request to the payor. All of this happens without the rep leaving Salesforce.

On the return path, PA status updates from seeQer, approved, denied, pending additional documentation, sync back into the Salesforce case in real time. The rep sees the current status inside the case they created.



RCM integration architecture: from fragmented systems to automated workflows

2. Pre-Submission Claims Validation

The claims problem was not just about wrong codes going out. It was about not knowing whether a claim would be accepted before submitting it.

We built seeQer's validation layer into the submission workflow. When a test is ordered, the system runs the CPT-ICD pairing through seeQer to check coverage, determine whether prior authorization is required, and surface the patient's estimated out-of-pocket cost, all before the patient receives care.

3. Proactive Denial Management via 835 Integration

Even with better validation upfront, denials still happen. Payors change rules, edge cases slip through, and documentation gets flagged on review.

We integrated 835 Electronic Remittance Advice files directly into the denial management workflow. When a payor processes a claim and decides not to pay, the integration picks up these files as they arrive, parses the denial reason code, and surfaces them in the system immediately. The workflow categorises denials by reason code and routes them into an appeals queue. The team reviews and submits rather than building an appeal from scratch, and nothing slips past the filing window.

The billing team went from reacting to rejections to preventing them. Error detection moved upstream, before the claim went out.

By the Numbers

Metric	Before	After
Revenue turnaround	Well beyond a 3-4-week window	**1-2 weeks**
Prior auth submissions	Inconsistent, manual	**Automated, standardised**
Denial rate	Elevated, compounding	**Materially reduced**
Rep workflow	Two systems, manual entry	**Single interface, real-time sync**
835 denial intake	Manual portal checks	**Automated parsing and routing**
Revenue visibility	Fragmented, unpredictable	**Real-time, single source of truth**
Scaling approach	Headcount	**System capacity**

The Impact

- **Prior authorization turnaround dropped significantly.** Standardised, automated PA submissions replaced inconsistent manual processes. Payor kickbacks from documentation gaps declined because the submission format became consistent across every rep and every case.

- **Claim denial rates decreased.** Pre-submission validation caught CPT-ICD mismatches and coverage gaps before claims went out. Denials that were preventable stopped happening.
- **The team could actually scale.** The same team handled significantly higher claim volumes because the system absorbed the repetitive, error-prone work. Throughput scaled without proportional headcount growth.
- **Revenue cycle visibility improved.** Real-time status sync into Salesforce gave leadership a single view across prior auth, claims, and denials. Revenue became predictable enough to plan around.
- **Appeals stopped slipping through.** Automated 835 intake and categorisation meant the appeals team worked a structured queue. Filing windows stopped being missed.

The lab stopped scaling people and started scaling systems. The integration layer made the existing team more effective than a team twice its size operating on fragmented workflows.

Beyond CareViso

NonStop has done similar work with Quadax. A genetic testing lab's billing submission process to Quadax was functional but fragile, with documents sometimes missing, files going out in incorrect formats, and errors not logged clearly enough for the RCM team to trust the output. We refactored the submission workflow to validate required documents before sending, auto-convert files to the correct format, and improve reliability and error logging end-to-end.

We do not sell boxed RCM products. We build and integrate systems that fit into the way a lab already operates. We have worked across the RCM ecosystem, CareViso, Quadax, and payor-specific portals, and our integration patterns are designed to be extensible. If the stack changes or a new payor comes on board with different requirements, the system adapts without a rebuild.

Want to see what this looks like for your lab?

nonstopio.com | Clinical Genomics & Life Sciences Engineering