

Contents

- Why RCM platforms still struggle despite strong features and workflows 3
- Where modern RCM platforms struggle across eligibility, authorization, claims and payments 4
- What fragmented healthcare data means for RCM platform builders and teams . . . 5
- Common RCM optimization strategies teams try first (and when they work) 6
- What actually needs to change to improve RCM outcomes at scale 7
- Why real-time connectivity across RCM stages is becoming essential 7
- How to assess whether your RCM platform needs deeper integration 7
- How revenue cycle API-driven integration supports each stage 8
- Next steps: how to test and implement RCM improvements without full system overhaul 9
- Using sandbox environments to try revenue cycle APIs before full integration . . . 9
- Evaluating RCM improvements based on outcomes, not features 10
- Scale RCM connectivity stage by stage 10
- Final takeaway: RCM performance depends on connectivity, not just capability 11

Why RCM platforms still struggle despite strong features and workflows

If you're building or improving an RCM platform, your success is measured by outcomes – lower denials, faster payments and better collections. Yet even well-designed platforms struggle.

1

According to the 2024 Revenue Cycle Denials Index by Optum, which analyzed 124 million hospital claim remits, **about 12% of claims submitted by U.S. hospitals are denied**, highlighting how common denials are across the revenue cycle.¹

2

In some payer environments, **first-pass denial rates climb above 30%**, underscoring how much performance can vary beyond national averages.²

3

Denials have continued to rise year over year, maintaining sustained operational strain and financial pressure for provider organizations.³

4

The financial impact is significant, with **denied claims accounting for more than \$48 billion in annual hospital revenue losses** across the healthcare system.⁴

These problems persist even when workflows are clean and teams are trained. That's because the issue rarely sits inside the RCM system itself. In most cases, failure emerges when information arrives too late or without verification, long after decisions have already been made.

Where modern RCM platforms struggle across eligibility, authorization, claims and payments

RCM performance doesn't break in one place – it breaks across stages, often quietly and progressively.



Eligibility errors at check-in lead to downstream claim denials

A patient may appear eligible during check-in, but without real-time validation, coverage gaps or exclusions go unnoticed. These errors often surface weeks later as denied claims, making them costly and difficult to trace back.



Prior authorization and benefit gaps slow down scheduling and care delivery

During scheduling, authorization and benefit checks are often delayed or incomplete. Teams rely on manual follow-ups, and approvals don't always arrive in time. Authorization-related issues contribute to more than 20% of claim denials, making front-end authorization one of the most common and costly failure points in the revenue cycle.⁵



Missing or incorrect data at checkout leads to preventable claim rework

At checkout, claims are created – but not always validated. Small errors like missing modifiers or incorrect coding slip through, only to be caught after submission. This leads to rework cycles that increase operational cost and delay payments.



Limited claim status visibility delays billing and increases A/R days

Once claims are submitted, many provider teams lack real-time visibility into claim status. Instead, they rely on manual follow-ups and periodic payer updates, which slows resolution and creates billing delays. As a result, reimbursement timelines commonly extend 6 to 8 weeks (45–60 days), contributing to higher days in accounts receivable and reduced cash flow.⁶



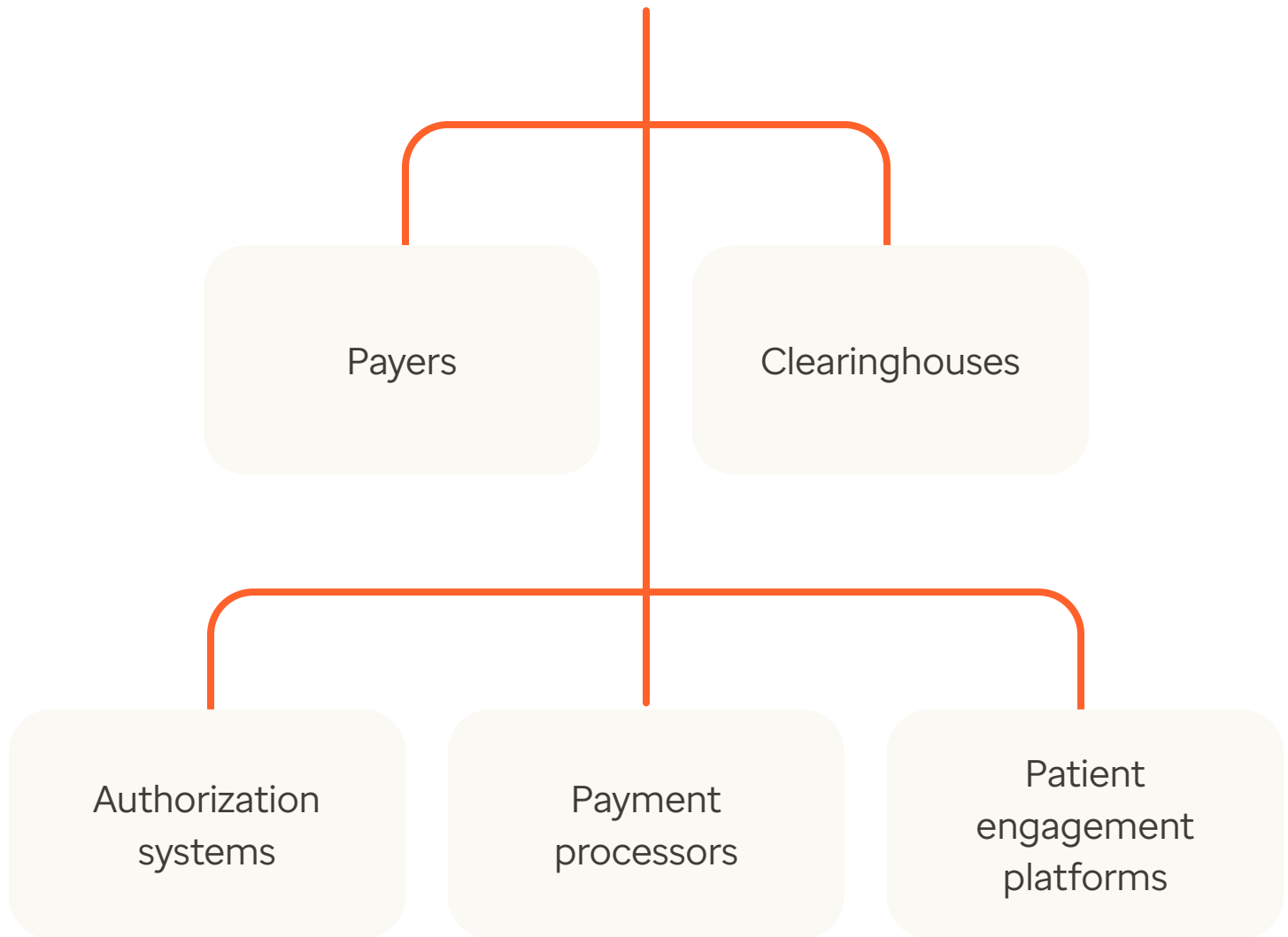
Manual reconciliation in payments increases cost and slows collections

At the payment stage, fragmentation between payments, remittance advice and billing systems often requires manual reconciliation. HFMA identifies manual workflows and limited automation as key drivers of higher cost-to-collect, particularly in payment posting, exception handling and follow-up activities. These inefficiencies slow collections and increase administrative burden, directly impacting revenue cycle performance.⁷

What fragmented healthcare data means for RCM platform builders and teams

Even if your platform is well designed, your outcomes depend on data you don't fully control.

That data flows across:



Your platform is not just executing workflows. It is reacting – constantly – to external inputs. When those inputs are delayed, incomplete or inconsistent, even the strongest internal design cannot fully compensate.

Common RCM optimization strategies teams try first (and when they work)

Before considering deeper changes, most teams try to optimize internally.

Adding more features to your RCM platform: when it works and when it fails

Adding features can improve usability and internal workflows.



It works well when the problem is within your system – such as missing functionality or inefficient user flows.



However, it fails when errors originate from external data sources, because features alone can't fix upstream inaccuracies.

Workflow optimization in RCM: when it improves efficiency and when it doesn't

Refining workflows can reduce inefficiencies and improve team productivity.



It works when processes are unclear or inconsistent.



But it falls short when delays are caused by external dependencies – like waiting for authorization or eligibility data.

Platform consolidation in revenue cycle management: benefits and limitations

Consolidating systems can simplify operations and reduce fragmentation.



It works when standardization is the primary objective.



However, dependencies on external systems remain. Those dependencies continue to drive latency and errors.

At this point, teams often reach an important conclusion:

This problem can't be solved by people, interfaces or internal optimization alone. Two systems must exchange data repeatedly, reliably and at scale. This is the moment when revenue cycle API-based integration stops being “complex” and starts becoming unavoidable.

What actually needs to change to improve RCM outcomes at scale

Across all these approaches, one pattern becomes clear: RCM performance doesn't break because of missing features – it breaks where systems need to connect. The issue isn't capability. It's connectivity and timing.

Why real-time connectivity across RCM stages is becoming essential

Modern RCM platforms are increasingly evaluated by how well they:

- Access external data in real time
- Validate information before errors occur
- Enable continuous visibility across workflows

This represents a shift away from optimizing isolated tasks toward enabling connected, end-to-end performance.

How to assess whether your RCM platform needs deeper integration

You can quickly identify whether connectivity is your main challenge by asking:

- Are denials happening despite clean workflows?
- Do issues originate from external systems?
- Are teams manually reconciling data across tools?
- Is there a delay between when data is needed and when it's available?

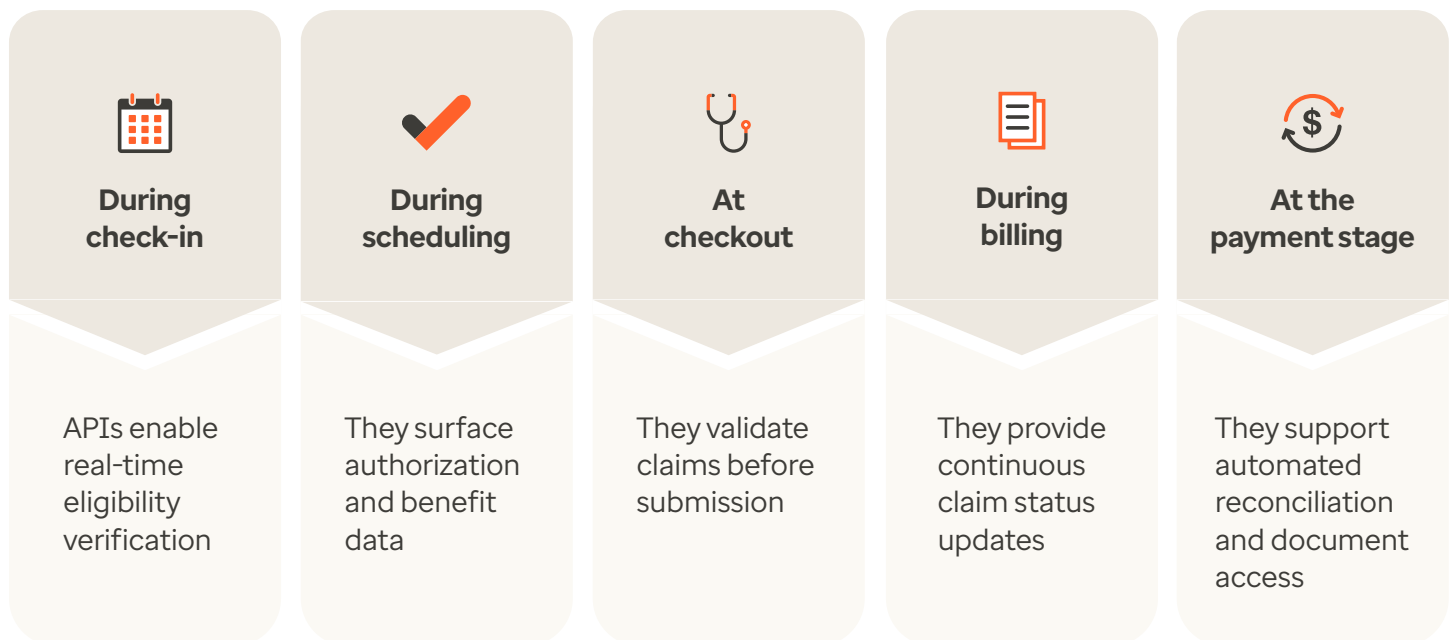
If the answer is yes, your challenge likely lies in integration – not internal design. This is typically the point where platform teams reevaluate build-versus-connect decisions.

How revenue cycle API-driven integration supports each stage

To enable real-time connectivity, platforms need a way to interact with external systems without friction.



This is where revenue cycle API-driven integration plays a role – as a layer that enhances your platform.



This does not eliminate payer variability or edge cases. It changes when and where those issues surface – earlier, when they are easier and less expensive to address.

Next steps: How to test and implement RCM improvements without full system overhaul

You don't need to rebuild your platform to explore this shift. Start small and focus on measurable impact. Deeper integration does not require replacing existing workflows or redesigning your system.

**Start with a high-impact use case in your revenue cycle.
Identify where the biggest friction exists:**



Eligibility
(if denials occur early)



Authorization
(if care is delayed)



Claims validation
(if rework is high)



Payments
(if reconciliation is manual)

Using sandbox environments to try revenue cycle APIs before full integration

Before making architectural changes, teams can validate integrations in controlled environments. Sandbox environments are designed for safe testing, using predefined requests and responses that reflect common scenarios. This helps teams understand how APIs behave before moving to production. As with most testing environments, sandbox interactions do not use live production data.

With modern [revenue cycle management solutions](#), teams can:



Access sandbox environments
for [revenue cycle APIs](#)



Explore workflows such as
eligibility checks, claims
validation and payments through
structured test scenarios



Evaluate integrations
without impacting existing
production systems



Key point

Not all APIs deliver the same outcomes. Choose a partner with proven experience in healthcare revenue cycle management and a track record of production-level API implementations. The right partner aligns integrations to your workflows and fits them into your existing systems.

Evaluating RCM improvements based on outcomes, not features

When testing any improvement, whether it's a workflow change, a platform update or an integration, the focus should stay on the outcome it has on day-to-day operations.

Many RCM teams introduce new capabilities and still see limited impact because implementation is completed, but performance does not materially change. Features may exist, yet the underlying issue continues to surface.

To make better decisions, improvements should be evaluated against clear and observable outcomes:



Real-time data availability

Check whether eligibility, authorization or claim status information is available at the exact moment it is needed within the workflow. This often depends on how well your platform connects with external data sources such as payers, clearinghouses and financial systems.



Reduction in manual intervention

Look at how much time teams spend calling payers, checking portals or reconciling data. Meaningful improvement usually comes from systems that can retrieve and exchange data without requiring manual follow-ups.



Prevention of errors upstream

Observe whether issues are identified earlier in the process. This includes eligibility gaps, missing documentation or incorrect coding. Early validation is often tied to how effectively your platform can access and act on the right data before submission.

Over time, the most valuable improvements are the ones that strengthen how your platform interacts with the broader revenue cycle ecosystem.

Scale RCM connectivity stage by stage

Once a specific improvement shows clear results, the next step is to expand in a controlled way. A stage-wise approach allows teams to move forward with confidence by keeping risk manageable and maintaining clarity on what is working. A practical way to scale looks like this:



Start with one high-friction stage

Focus on the part of the revenue cycle where challenges are most visible. For example, eligibility when denials start early, or claims validation when rework is frequent. This helps demonstrate value quickly and identify where better data access can improve outcomes.



Extend to adjacent stages where dependencies exist

Improvements in one stage often influence others. Stronger eligibility checks can support better authorization outcomes and cleaner claims. Expanding into connected areas helps create continuity across workflows.



Build a progressively connected revenue cycle

As more stages improve, the overall system becomes more consistent. Data flows more reliably across systems, visibility improves and teams spend less time managing exceptions. This is typically achieved by strengthening how the platform integrates with external services across each stage.

This approach supports steady progress. It allows teams to improve performance while maintaining flexibility in how they extend their platform capabilities over time.

Final takeaway: RCM performance depends on connectivity, not just capability

Whether you're building your own RCM platform, extending an existing one or integrating multiple systems — your ability to deliver outcomes depends on how well each stage of your revenue cycle connects to the broader healthcare ecosystem.

Sources

1. Optum AI Marketplace. [The Optum 2024 Revenue Cycle Denials Index](#). Accessed April 10, 2026.
2. Kaiser Family Foundation. [Claims denials and appeals in ACA marketplace plans in 2024](#). Accessed April 10, 2026.
3. Healthcare Finance News. [Claims denials are on the rise, complicating revenue collection](#). Accessed April 10, 2026.
4. Healthcare Finance News. [Hospital net revenue leakage increases due to denied claims](#). Accessed April 10, 2026.
5. Healthcare Finance News. [Payer denials and prior authorization delays are top revenue cycle concerns](#). Accessed April 10, 2026.
6. Healthcare Finance News. [Revenue cycle challenged by low collection rates and high denials](#). Accessed April 10, 2026.
7. Healthcare Financial Management Association. [Cost to collect better practices](#). Accessed April 10, 2026.