

ML LABS INTELLIGENCE

Engineering Bidirectional HL7 for AI Result Delivery

ML LABS built the HL7 messaging layer that delivers AI diagnostic results to hospital EHRs — with per-organization configuration, billing code flow, and duplicate prevention across every deployment.

CASE STUDY

Author Omar Trejo

Date 2026-02-21

ML LABS

mllabs.com/intel/hl7-ai-result-delivery-system

A healthcare AI platform had accurate diagnostic results but no way to get them back into the hospitals ordering them. The platform needed bidirectional HL7 messaging — receiving orders from EHR systems and delivering AI-generated results in a format each hospital would accept without rejection. This is not integration setup. It is production-grade message engineering with compliance at every level, diagnosis code handling for billing, delivery timing that prevents duplicates, and per-organization configuration that makes one codebase serve hospitals with different HL7 requirements.

ML LABS engineered this entire messaging layer. The engagement covered order message handling with billing code flow, result delivery with the clinical measurements and coding that downstream systems depend on, per-organization configuration that eliminates hardcoded defaults, delivery state tracking across preliminary and final result paths, and corrections driven by direct EHR vendor feedback. The [companion article on EHR and FHIR integration](#) covers the integration setup layer — authentication, FHIR, sandbox access. This article covers what happens after setup: building the actual messages.

The Problem

HL7 v2 messaging remains the dominant transport for clinical orders and results in hospital systems. [Standard clinical coding systems](#) provide observation identifiers for HL7 message segments. But the standard is a starting point. The actual message that any given EHR will accept depends on site-specific configuration, vendor-specific requirements, and organizational policies that vary across every hospital deployment.

The platform needed to handle multiple order workflow types, result delivery with billing codes that affect downstream revenue cycle processing, and per-organization mappings that change across every hospital. Discovering these variations one hospital at a time — through rejected messages and support tickets — is slow, expensive, and erodes clinical trust in the AI system.

Order Messages with Billing Code Flow

When the AI platform communicates order-level information back to the EHR, every message must match the originating order precisely so the EHR can file it correctly. ML LABS built the complete order workflow with full test coverage across the message structure.

Order Tracking

Return messages must reference the original order identifiers exactly as the EHR assigned them. A mismatch means the result arrives but cannot be filed — it enters a limbo queue requiring manual intervention from the hospital's interface team. ML LABS resolved cases where EHR configurations validate order references at multiple points in the message independently.

Billing Code Flow

Diagnosis codes selected during clinician review carry direct billing implications — they must flow through the HL7 message so the EHR can associate them with the order for revenue cycle processing. ML LABS built a billing code selection UI that enables clinicians to attach ICD codes during review, with those codes propagating into the outbound message.

The billing code requirements are deceptively specific — each organization requires different code subsets, supports different levels of diagnostic complexity (primary, secondary, tertiary), and enforces different formatting rules. ML LABS built the system to handle any diagnostic complexity without per-hospital code changes, iterating the code catalog as new hospitals onboarded with different requirements.

Delivering AI Results

The result message is the core delivery mechanism. It carries AI-generated clinical findings back to the EHR in a structured format the ordering clinician can review within their normal workflow.

Clinical Measurements and Coding

ML LABS built bidirectional HL7 messaging that delivers AI results back to the EHR with the clinical measurements and billing codes that downstream systems depend on. For the AI ECG platform, this includes heart rate, rhythm classification, RR interval, and axis measurements — each coded so the EHR files, displays, and trends them correctly.

Compliance at the message level is where HL7 integration moves from "works in the sandbox" to "works in production." The specification tells you which fields exist. The EHR tells you which values it will accept — and that answer changes per site.

Getting the clinical coding right matters beyond filing — an incorrect code can cause a result to appear in the wrong chart section or fail to trigger clinical decision support rules. ML LABS corrected coding after specification review revealed that technically valid codes were operationally incorrect for the target EHR configurations.

Patient and Provider Validation

Patient identifier and ordering provider fields are among the most strictly validated elements in a result message. They must match the EHR's internal formats exactly.

ML LABS fixed both patient identifier and ordering provider formatting based on direct feedback from the EHR integration team. These corrections are the kind of issue that only surfaces through iterative testing against the actual target EHR — the specification documents which fields exist, but the EHR decides which values it will accept.

Delivery Timing and Duplicate Prevention

The platform supports multiple result delivery workflows — both responses to EHR orders and results initiated by the AI platform — including preliminary result delivery through web upload paths.

ML LABS built delivery coordination that prevents duplicate sends across all result pathways. When multiple delivery triggers exist for the same study, the system ensures exactly one result reaches the EHR. Per-organization configuration controls which result types are enabled — some organizations want early preliminary visibility while others require formal sign-off before any result delivery. These settings are configurable per site without code changes.

Per-Organization Configuration

One codebase serving multiple hospitals means no hardcoded defaults. Every HL7 field that varies across organizations must be configurable per site with validation that fails loudly when required configuration is missing.

What ML LABS Built

ML LABS implemented per-organization HL7 configuration that externalizes every site-specific value — facility identifiers, application names, patient identifier formats, provider reference schemes, diagnosis code subsets, and authentication bindings. The implementation explicitly requires all fields per organization and removed hardcoded defaults that had accumulated during initial single-hospital development.

Each hospital's integration authenticates with its own credentials bound to its organizational context — messages for Hospital A cannot be authenticated against Hospital B's endpoint, even if a credential is compromised.

Why Hardcoded Defaults Break

The common early-stage shortcut is hardcoding values that work for the first customer. This creates technical debt that compounds with every new hospital — each site has different requirements for identifier formats, diagnosis code support, and provider reference schemes. ML LABS built validation that rejects message construction when required per-org configuration is missing, preventing silent misconfiguration that would otherwise surface as rejected messages in production.

Boundary Condition

Bidirectional HL7 v2 messaging is the right approach when the target EHR uses HL7 v2 as its primary integration interface — which remains the majority of US hospital systems. Some newer deployments support FHIR-based result delivery with different tradeoffs: richer data models, query-based access, and potentially less per-site variance.

If the target integration is exclusively FHIR-based, the segment-level engineering described here does not apply directly. In practice, most production healthcare AI deployments require both — FHIR for data access and HL7 v2 for order and result messaging.

First Steps

1. **Define the message catalog before writing code.** List every order and result message type required. Get the target EHR's integration specification — not the generic HL7 standard — and map requirements per organization.
2. **Build per-org configuration from the start.** Every value that could vary across hospital deployments should be configurable per organization with validation that fails loudly when required configuration is missing. Retrofitting per-org configuration into a hardcoded implementation always costs more than building it correctly the first time.
3. **Implement delivery coordination for every result pathway.** Preventing duplicate sends across preliminary, final, and document delivery pathways is not optional. Double sends create clinical confusion and erode trust in the AI system faster than any other integration failure mode.

Practical Solution Pattern

Engineer the HL7 messaging layer as a per-organization configurable system with mandatory validation against each site's EHR requirements and delivery coordination that prevents duplicates across all result pathways. Externalize every site-specific value into per-org configuration from the first deployment. Build billing code flow with support for multiple diagnoses, and coordinate delivery triggers so no study sends duplicate results.

This works because the primary failure mode in HL7 integration is not message construction logic — it is site-specific field values that silently break when a second hospital has different requirements than the first. Organizations that get the per-org architecture right on the first implementation add new hospital sites as configuration tasks rather than code changes. Deep expertise paired with AI-augmented execution now compresses what would otherwise be an extended cycle of rejected messages, vendor feedback loops, and field-by-field debugging into a tight delivery that reaches production with compliance verified. If the HL7 messaging requirements are defined and the constraint is production-quality engineering, **AI Workflow Integration** is the direct build path.

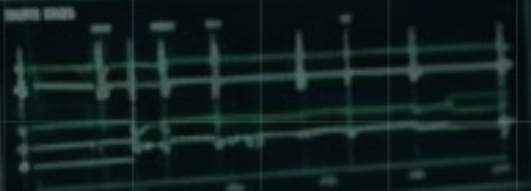
References

1. HL7 International. **FHIR R5 Specification**. HL7 International.
2. Regenstrief Institute. **LOINC and Other Standards**. LOINC.
3. Rhapsody Health. **HL7 Observation Result (ORU) Message**. Rhapsody.
4. Rhapsody Health. **HL7 ORM Message**. Rhapsody.
5. Wang, F., Beecy, A. **Implementing AI Models in Clinical Workflows: A Roadmap**. *BMJ Evidence Based Medicine*, 2025.
6. ONC. **Health Data, Technology, and Interoperability: Certification Program Updates**. HealthIT.gov, 2024.
7. 6B Health. **Integrating with EHR Systems Using HL7 V2**. 6B Health, 2024.



MEDICAL DATA PACKETS

PACKET ID	DATE	STATUS
000001	2023-01-01	ACTIVE
000002	2023-01-02	ACTIVE
000003	2023-01-03	ACTIVE
000004	2023-01-04	ACTIVE
000005	2023-01-05	ACTIVE
000006	2023-01-06	ACTIVE
000007	2023-01-07	ACTIVE
000008	2023-01-08	ACTIVE
000009	2023-01-09	ACTIVE
000010	2023-01-10	ACTIVE
000011	2023-01-11	ACTIVE
000012	2023-01-12	ACTIVE
000013	2023-01-13	ACTIVE
000014	2023-01-14	ACTIVE
000015	2023-01-15	ACTIVE
000016	2023-01-16	ACTIVE
000017	2023-01-17	ACTIVE
000018	2023-01-18	ACTIVE
000019	2023-01-19	ACTIVE
000020	2023-01-20	ACTIVE



2023-01-20 REPORT

Summary of data for the period 2023-01-20. The report includes a list of data points and their corresponding values.

Item	Value
Item 1	100
Item 2	200
Item 3	300
Item 4	400
Item 5	500

ML LABS

Custom AI Systems for High-Value Workflows

mllabs.com