// ARTICLE · PUBLISHED 24 SEPTEMBER 2026

Epic Migration Planning: Twelve Thousand Questions, One Date

An Epic program gets planned as a build, and it succeeds or fails on how fast its decisions get made.

Jason Newell · MAX Research Collective · Article ART-HLC-010 · Published 24 September 2026
Licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0), https://creativecommons.org/licenses/by/4.0/
Original: https://www.maxresearchcollective.com/articles/epic-migration-planning.html
Cite as: Newell, J. (2026). Epic Migration Planning: Twelve Thousand Questions, One Date. MAX Research Collective, ART-HLC-010. https://www.maxresearchcollective.com/articles/epic-migration-planning.html

Most Epic migration plans start with the build: the modules, the teams, the environments and a go-live date. That's the visible work, and it's the part boards ask about. The thing that decides whether the date holds sits somewhere else. It's the number of decisions the program has to make, and how fast it can make them.

A Norwegian regional health authority planned its Epic program around exactly that. Its planners expected that Epic "will probably raise something like 8,000 to 12,000 questions when they start the implementation" (Hertzum and Ellingsen, 2019). Each one needs an answer from someone with the authority to give it. Which order set does cardiology use? Who's allowed to see a behavioural-health note? What happens to a referral that arrives for a clinic that closed last year?

- mortenhertzum.dk · IJMI2019

Put that number on a calendar. A program of about a dozen hospitals usually spends twelve to sixteen months in discovery and build, which is roughly 260 to 350 working days. Spread the questions across them and the program has to settle somewhere between 23 and 46 decisions every working day. The go-live date won't move to make room; in most programs the contract fixed it before the first question showed up.

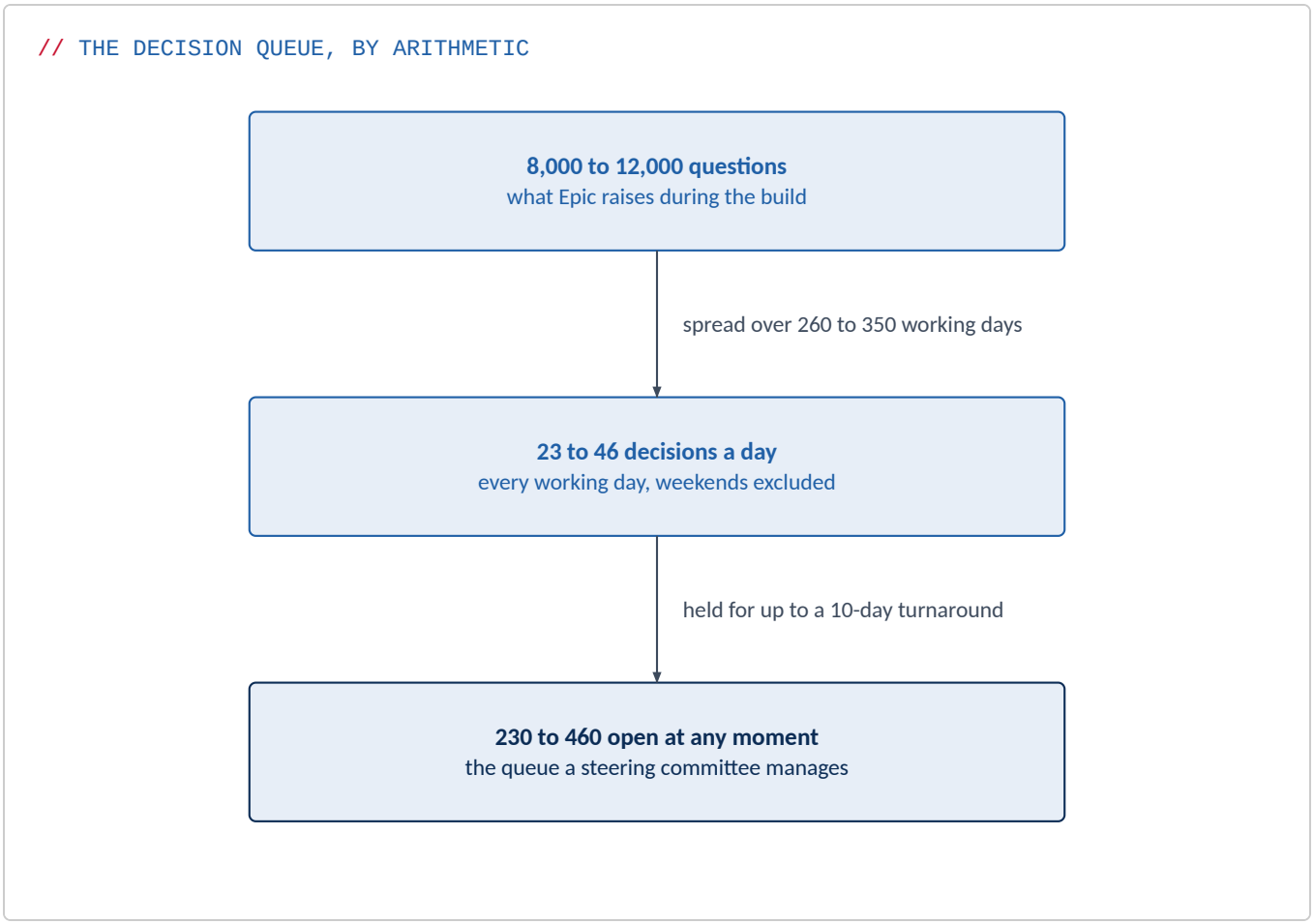


Figure 1. Twelve thousand questions spread over a fixed build window leave hundreds open at any moment, even with a ten-day turnaround.

Decisions are the critical path

Program plans are usually drawn as workstreams: clinical build, revenue cycle, integration, training and technical infrastructure. Each one has a lead, a budget and a bar on the Gantt chart. A decision that falls between two workstreams has none of those, so it sits and waits for someone to claim it.

Here's a simple example. Cardiology wants its own order set; the enterprise standard says everyone uses one. The clinical build team can't decide it, and neither can the physicians' workgroup on its own. Until someone with authority rules, the build for that order set stalls, and so do the tests that depend on it.

Norway's answer was a mandate. Any question raised had to be decided within ten working days, by a structure that gave line managers the authority to decide it. The

same study shows what happened in programs without that mandate. Committees stalled, and in the areas left open, Epic's standard configuration (its Foundation System) became the answer by default. Not deciding turned out to be a decision. Nobody made it on purpose, and the clinicians who inherited it were never asked.

Ten days sounds quick. Run the numbers, though. At 23 to 46 new questions a day, a ten-day turnaround still leaves 230 to 460 questions open at any moment. That's the queue a steering committee is really managing, and it almost never shows up on the agenda. Adventist Health set up 77 workgroups for its program (This Week Health). Spread the open questions across that many groups and each one carries three to six at a time, which a weekly meeting can clear.

- thisweekhealth.com · its on us cdio jennifer stemmler on the roles that ownership…

Holding the ecosystem around an Epic implementation at a large multi-state integrated payer-provider system meant planning data, integration, cloud and identity work against decisions made inside Epic's build. Other teams made those decisions, on their own calendar. The ecosystem plan logged each decision once it was written down and treated it as settled from then on. Testing proved that wrong. A decision with no named owner reopened the first time it cost someone something, and every reopened decision moved an interface, an extract or a provisioning rule that had already been built on it. That was an error in how decisions were tracked, and the engineering around them was fine. The register needed an owner and a date on every decision the ecosystem depended on — and it had neither.

What has to exist before go-live

Go-live is a single weekend. The architecture around it runs on its own schedule, and the most important scoping conversation on the program is about where to draw a line: what has to be finished before the date, and what can safely follow it. Draw the line too late and the critical path fills with work that could have waited. Draw it too early and something the hospital depends on gets pushed past go-live.

MUST BE COMPLETE BEFORE GO-LIVE	CAN FOLLOW GO-LIVE, WITH DATES
Identity and access: user and provider accounts, single sign-on, badge tap	Access reviews, and handling staff who change roles or leave
Every clinical interface live and tested at full production volume	Daily checks that every message sent was also received, for the first 90 days
Network connections to Epic, including firewall and allowlist changes	Modernizing the analytics platform and adding real-time data feeds
Clarity and Caboodle running, with day-one regulatory and billing reports checked	FHIR integration beyond the regulatory minimum
All the non-production copies of Epic the program needs	Any move to Epic's cloud analytics platform

Table 1. The left column is the critical path, and nearly every item on it is owned by someone outside the Epic build team.

Reporting belongs on the left, and it's the item most often argued onto the right. Clarity is Epic's overnight reporting copy of its live database, and Caboodle is the data warehouse built on top of it. The best-documented regret in the research is a program that treated reporting as something to finish after go-live. Sixteen months later, nearly half of its 750 reports had been pulled as invalid, and baseline data was missing for 20 of 26 productivity measures. Two years on, the national audit office couldn't tell whether productivity had recovered or whether the reports were simply wrong (Hertzum and Ellingsen, 2019). Nobody could say which.

- mortenhertzum.dk · IJMI2019

Cloud work has a proven order too. KLAS found that disaster recovery was the first Epic workload most organizations moved to public cloud, at 78 percent of those interviewed (KLAS). Starting with recovery makes sense. It gives the hospital something real without putting go-live at risk.

- pages.awscloud.com · 24 GLOBAL en US klas research other ardm klas epic in the cloud…

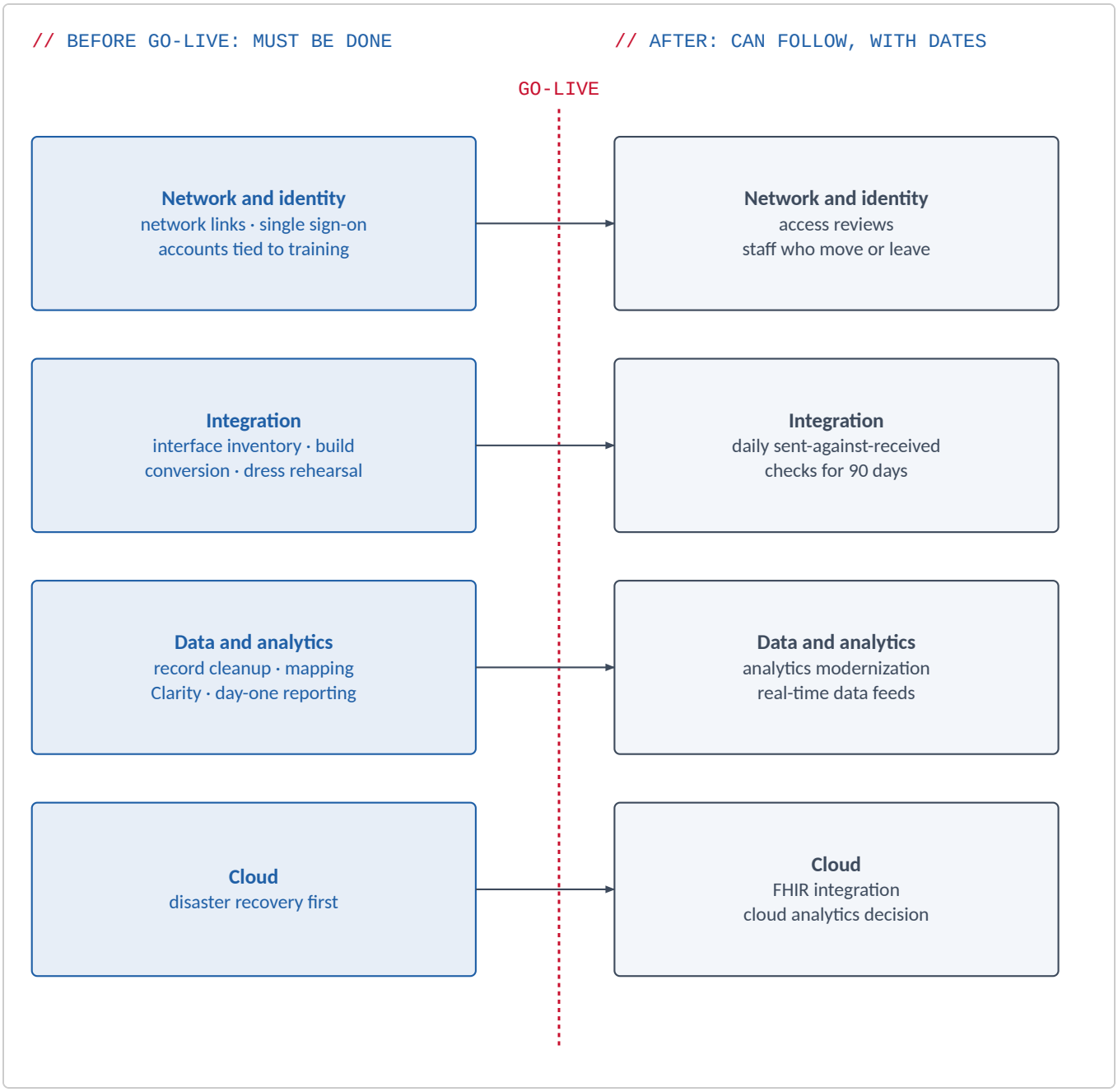


Figure 2. Identity, integration and day-one reporting have to be finished before the date, while analytics modernization and most cloud work can follow.

How long each phase takes

Epic's own guidance is that smaller systems commonly reach go-live in 9 to 12 months, and large ones reach their first go-live in 16 to 18 months, with later waves every six months (Becker's). Real programs start the clock in different places, which makes them hard to compare. WMCHealth took about 20 months from kickoff to go-live, Intermountain about 24 from its announcement, and Lewisham and Greenwich

about 13 from contract signature. A benchmark that doesn't say which clock it used isn't much help.

- beckershospitalreview.com · how long epic ehr implementations take

PHASE	TYPICAL DURATION	WHAT IT DEPENDS ON
Discovery, capturing how work is done today	4 to 6 months	Governance in place, with a decision mandate
Build and data mapping	8 to 10 months	Decisions arriving as fast as the build needs them
Testing, from single pieces to the whole system, with at least three practice data loads	6 to 9 months, overlapping build	Build mostly complete; mapping finished
Technical dress rehearsal	60 to 90 days before go-live	Every device installed
Role-based training	8 to 12 weeks	A build stable enough to train on
Appointment and case conversion	One weekend, about two weeks before go-live	Scheduling build complete
At-the-elbow support	2 to 4 weeks, then 30 to 60 days of heightened support	Staffing matched to how busy each unit is
Productivity back to normal	6 to 12 weeks	Training that stuck
Cash back to normal	120 to 180 days	Revenue cycle funded past go-live

Table 2. Typical phase durations for a program of about a dozen hospitals, drawn from the published programs and vendor guidance cited in this section.

Each phase leans on the one before it. Build has to be mostly done before mapping can finish, and mapping has to finish before testing means anything. So a month lost in build shows up as a month lost in testing, and it keeps rolling forward.

The technical dress rehearsal deserves its own line in the budget. It checks every workstation, printer, scanner, badge reader and sign-on path in every clinical area. Workstation and printer problems are consistently among the top issues at go-live (HCTec). A nurse who can't print a wristband at 7 a.m. on day one doesn't care how well the order sets were built. It's dull work, and it's where go-lives are won or lost.

- hctec.com · technical dress rehearsal

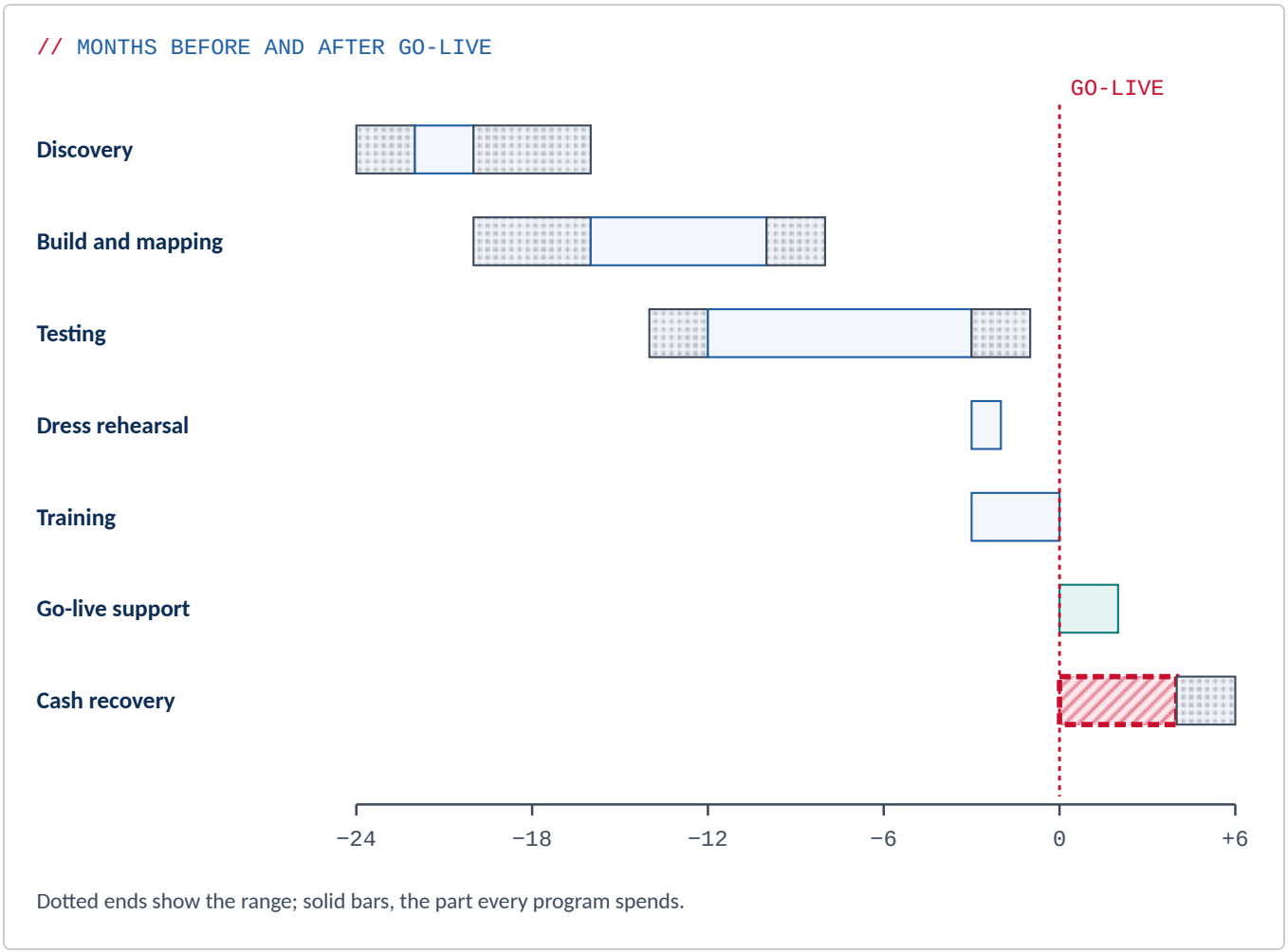


Figure 3. The phases overlap by design, so a slip in build pushes every phase to its right.

What it takes in people

Epic's staffing guidance goes to customers under NDA and isn't public. The organization signing the contract already has that document, though. Asking for it, along with how the program is tracking against it, is a better first week than chasing outside benchmarks.

The best public reference is a 16-hospital EHR and revenue-cycle implementation documented by Impact Advisors. Its core project team was 350 people, including 68 from Epic, backed by 3,300 subject-matter experts across 62 workgroups (Impact Advisors). That's about one core team member for every 105 employees, and roughly ten clinical experts behind each of them. That program ran in waves over five years. A big bang squeezes the same build and testing into one go-live, which usually means a peak team at least as large over a shorter stretch. Boards rarely expect that.

- impact-advisors.com · an ehr and rev cycle system implementation

Go-live itself is a staffing event, and most of the people come from the hospital's own rosters. BJC's first wave pulled 552 super users — front-line staff trained in depth — off their normal shifts. It also brought in about 154 contracted at-the-elbow staff and ran a command center of more than 200 people at peak, around the clock (BJC). A staffing firm's published ratios show how support scales with how busy a unit is.

- usermanual.wiki · help

SETTING	DAYS 1 TO 14	WEEK 3 ONWARD
Emergency department, ICU, operating room	1 per 3 to 4 clinicians	1 per 10 to 15
Medical-surgical and telemetry	1 per 5 to 7	1 per 10 to 15
Ambulatory clinics	1 per 6 to 10	1 per 10 to 15

Table 3. At-the-elbow support ratios from a staffing firm's guidance, best treated as planning figures rather than measured outcomes (Trusted Talent).

- trustedtalent.com · epic staffing strategy

IT headcount is the surprise. After go-live, the IT department is about the same size. The University of Vermont Health Network filed the most detailed cost breakdown in the public record, a certificate of need its state regulator required. It projected about 85 legacy positions going away and 63 Epic support positions being created, a net reduction of 22. The same filing put outside staffing at $36.4 million, which was 2.5 times the $14.3 million Epic licence (Green Mountain Care Board). Outside help is normal on the infrastructure side as well. Of the organizations that moved Epic workloads to public cloud, 75 percent hired outside firms to do it (KLAS). Budget for that help at the start; hired in a hurry mid-program, it costs more and comes on the vendor's terms.

- gmcboard.vermont.gov · CON application Epic EHR 1 3 17 with covers and exhibits
- pages.awscloud.com · 24 GLOBAL en US klas research other ardm klas epic in the cloud...

Where programs are known to break

Every failure below has a published case behind it and a cost that lands on a specific budget. Most of them also belong to someone outside the Epic build team, which is exactly why they keep happening.

GOTCHA	WHAT IT COSTS	WHO SHOULD OWN IT
Reporting pushed past go-live	Half of 750 reports invalid 16 months later, and no way to tell whether productivity recovered	Chief data or analytics officer
Location and provider records with no routing check	At the VA, 11,000 orders silently went to an unknown queue: 149 adverse events and about 16 months to fix	Integration lead, backed by patient safety
Patient record cleanup scoped only from known systems	One program found 22 source systems instead of 5, and the 17 unknown ones held about a third of all patients	Health information management
Old software contracts left to auto-renew	Retirements slip by a year, and the savings in the business case slip with them	Procurement, working from the application inventory
Conversion testing and system testing competing for the same environments	Both need full-size data in the final months, and a big bang stacks them on top of each other	Technical infrastructure, with one shared calendar
API fees assumed to be free	$4,500 to $47,000 a year per production instance for the core FHIR bundle, billed to the health system	Integration architecture and finance
The network link between Epic's hosting and the hospital's own systems	Firewall and port changes were the top integration surprise in KLAS's cloud study, and each one waits on an outside party	Network and infrastructure, as its own workstream
Data extracts sized for an average day	Penn's Clarity load skips Sunday, so Monday's load carries two days of data and breaks any job sized for one	The team that owns the data warehouse

Newell, J. (2026). MAX Research Collective, ART-HLC-010. CC BY 4.0

GOTCHA	WHAT IT COSTS	WHO SHOULD OWN IT
Quarterly Epic upgrades with nobody reading the change log	Integrations stop working after an upgrade, in Epic's words "with no indication of what changed"	A named owner of an integration registry

Table 4. Nine documented failure points, each with its cost and the owner it usually lacks.

Sources: Hertzum and Ellingsen; VA Office of Inspector General; e4health; Harmony Healthcare IT; Epic; KLAS; Penn Medicine.

- mortenhertzum.dk · IJMI2019
- vaoig.gov · new electronic health records unknown queue caused multiple
- e4.health · healthcare system leverages e4healths patient identity management…
- harmonyhit.com · taking control of application sprawl a strategic approach for health…
- epic.com
- pages.awscloud.com · 24 GLOBAL en US klas research other ardm klas epic in the cloud…
- med.upenn.edu · epic clarity data warehousing

Two of those deserve more than a row. The VA case is worth knowing by heart, because its cause was so ordinary. Clinicians could pick a facility location from a dropdown that didn't properly exist in the new system. Orders placed against that location went nowhere, and the ordering provider's screen still said the order had been placed. Location and provider records are usually rebuilt during a migration instead of copied, which makes them some of the riskiest data on the program. The fix is cheap. Count orders placed against orders received in each department every day for the first 90 days, and raise an alarm on any gap, however small, because a silent queue always starts small.

The second is data mapping. On real conversions, spelling variants, grammar and typos have made up roughly 90 percent of the work of mapping old values to new

ones (Hayes Management). Picture a lab test entered as ”HbA1c”, ”HGBA1C” and ”Hemoglobin A1c” in three clinics, each needing the same mapping. New variants keep turning up until the old system is switched off, so mapping runs as a loop right up to cutover, and it needs staff for the whole stretch.

- info.hayesmanagement.com · data conversion best practices

The money, briefly

The money has its own article, ”EHR Migration Economics: The Cash Trough Nobody Budgeted,” so two numbers will do here. The Epic contract is typically 25 to 45 percent of what a program costs; people, hosting, training, interfaces and running two systems side by side make up the rest. Cash takes 120 to 180 days to get back to normal after go-live. A liquidity plan has to cover four to six months past the date, and most budgets carry six weeks. A timeline that ends at go-live leaves out the months a board will judge it on.

Counting the questions

A technology leader signing off on an Epic plan usually reviews the builders, the environments, the interfaces and the dates. All of that is real work, and every piece of it waits on decisions.

Checking the decision queue takes one morning and three questions for the program office. How many design decisions are open today? How many were closed last week? How long has the oldest one been waiting? A program that can answer has a queue it can manage. One that can’t is carrying the Norwegian number without knowing it, and it will find out in testing, when every open question arrives at once with a defect attached.

Glossary

TERM	DEFINITION
Allowlist	An allowlist is the set of network addresses allowed to connect to a system. On Epic-hosted programs, changing it takes a ticket and an outside party's lead time.
At-the-elbow support	These are trained staff who stand beside clinicians during the first weeks after go-live.
Big bang	A big bang moves every site onto the new system in one cutover. It avoids building bridges between old and new systems, and it puts all the risk into a single weekend.
Caboodle	Caboodle is Epic's data warehouse, loaded from Clarity and shaped for analysis.
Clarity	Clarity is Epic's reporting database, a copy of the live system loaded overnight. Anything read from it is at least a day old.
Cutover	Cutover is the period, usually a weekend, when the old system stops and the new one starts.
Enterprise master patient index (EMPI)	The EMPI links every record that belongs to the same patient across systems. Duplicates carried into the new system multiply instead of disappearing.
FHIR	Fast Healthcare Interoperability Resources is the web standard Epic uses for reading and writing clinical data. Epic bills health systems for its core FHIR bundle per production instance.
Foundation System	This is Epic's standard configuration. A question a program leaves undecided usually ends up with the Foundation answer.
Hypercare	Hypercare is the period of heightened support after go-live, typically 30 to 60 days, before issues move to the normal help desk.

Newell, J. (2026). MAX Research Collective, ART-HLC-010. CC BY 4.0

TERM	DEFINITION
Integration registry	An integration registry lists every interface with its owner, the Epic features it uses and how critical it is. Epic writes its change log in terms of exactly those details.
Master data	Master data is the reference information other records point to, such as locations, departments and providers. When it's wrong, orders go nowhere.
Non-production environment	These are the copies of Epic used for building, testing, training and practising conversion. Large programs run about twenty of them.
Subject-matter expert (SME)	An SME is a clinician or operational specialist who knows how the work is actually done. Programs draw on thousands of them for design decisions.
Super user	A super user is a front-line staff member trained in depth who supports colleagues on the unit during and after go-live.
Technical dress rehearsal (TDR)	A TDR tests every device, printer, badge reader and sign-on path in every clinical area, usually 60 to 90 days before go-live.

HEALTHCARE ART-HLC-010

// ABOUT THE AUTHOR

Jason Newell leads organizations and people in technology, and is a researcher, deep technologist and builder — from a CRAY to agentic AI. He founded MAX Research Collective to build AI for regulated industries that runs where its owner has control, with healthcare as its strongest focus. About Jason → Connect on LinkedIn →

Newell, J. (2026). MAX Research Collective, ART-HLC-010. CC BY 4.0