

Escaping the Software Graveyard: Rationalize, Integrate, and Budget for AI

Spark Healthcare Advisory · Oct 22, 2026

SPARK HEALTHCARE ADVISORY · INSIGHTS

**Know what you own
before you buy what's next.**

Rationalize · Integrate · Budget for AI

Spark Healthcare Advisory · Insights

In August 2026, 43% of hospitals were operating in the red, according to Kaufman Hall's National Hospital Flash Report ([Becker's Hospital Review](#)). When margins are that thin, every dollar spent on software nobody uses is a dollar not spent on staff or patients.

Most health systems are carrying more of that software than they realize. Walk through almost any organization and you'll find a quiet graveyard of platforms that everyone stopped opening six months after launch, still renewing every year.

The waste is easy to underestimate. Across industries, the average organization uses only 54% of the SaaS licenses it pays for and wastes \$19.8 million a year on unused licenses, according to Zylo's 2026 SaaS Management Index. For organizations with more than 10,000 employees, the average is \$80.6 million ([Zylo](#)).

Health system leaders know it's a problem. In a CHIME survey, 76% of CIOs called application rationalization critical, but only 1 in 5 had a program in place ([Becker's Hospital Review](#)).

This piece covers how the graveyard grows, how to clean it up, how to make what stays work together, and how to keep the next wave of AI tools from joining it.

How the graveyard grows

Nobody sets out to buy software that sits unused. It builds up through a few common habits:

- **Roadmaps that end on launch day.** Plans built on best-case conditions, with no budget or owner for what comes after. When staffing gets tight or the data is messy, the tool quietly stops being used.
- **Layering instead of integrating.** New platforms get stacked on top of legacy systems with no plan for how they fit together. Staff end up managing systems instead of workflows.
- **Department-by-department buying.** One team buys a tool to solve its problem. Another team buys something that overlaps. Nobody sees the whole picture.
- **No owner after go-live.** The project team moves on, and nobody is accountable for whether the tool is still delivering.
- **Auto-renewals.** Contracts renew because nobody checked usage before the notice date.

The result isn't modernization. It's administrative drag: more logins, more interfaces to maintain, more security exposure, and more cost.

What it really costs to keep an app

The license fee is only the visible part. Every app also carries:

- **Interfaces.** Each connection to the EHR, the ERP, or another system has to be built, monitored, and fixed whenever either side upgrades.
- **Security exposure.** Every app adds accounts to manage, a vendor with access to your data, and a system to patch.
- **Staff time.** Analysts who maintain it, help desk calls, and training for every new hire.
- **Upgrade testing.** Each release of a core system means retesting everything connected to it.
- **Clinician attention.** More logins and more places to look. When Providence cut its portfolio, a leader there said the goal was to "lower the cognitive burden of our caregivers" by removing duplicate tools ([Becker's Hospital Review](#)).

Clean it up: a practical approach

Application rationalization sounds like a big consulting exercise. It doesn't have to be. The goal is simple: know what you have, what it costs, who uses it, and whether you still need it.

1. **Build the inventory.** Every application, its business owner, its technical owner, what it does, and what it connects to. Include the tools departments bought on their own.
2. **Add cost and usage.** License, hosting, support, and interface costs for each app, next to real usage: active users, logins, transactions. Usage data usually tells a clearer story than surveys.
3. **Map the overlap.** Group apps by what they do. Three tools doing the same job is a consolidation candidate. A function your core platform already covers is a retirement candidate.
4. **Decide each app's fate.** Keep, consolidate, replace, or retire. Make the call with the people who use it, not just IT.
5. **Plan the data.** Decide what gets migrated, what gets archived for legal and clinical lookback, and how people will find it later.
6. **Time it to the contracts.** Line up retirements with renewal and termination dates so the savings start on schedule.
7. **Report the savings.** Track retired costs and redirected spend, and share the results. Visible wins make the next round easier.

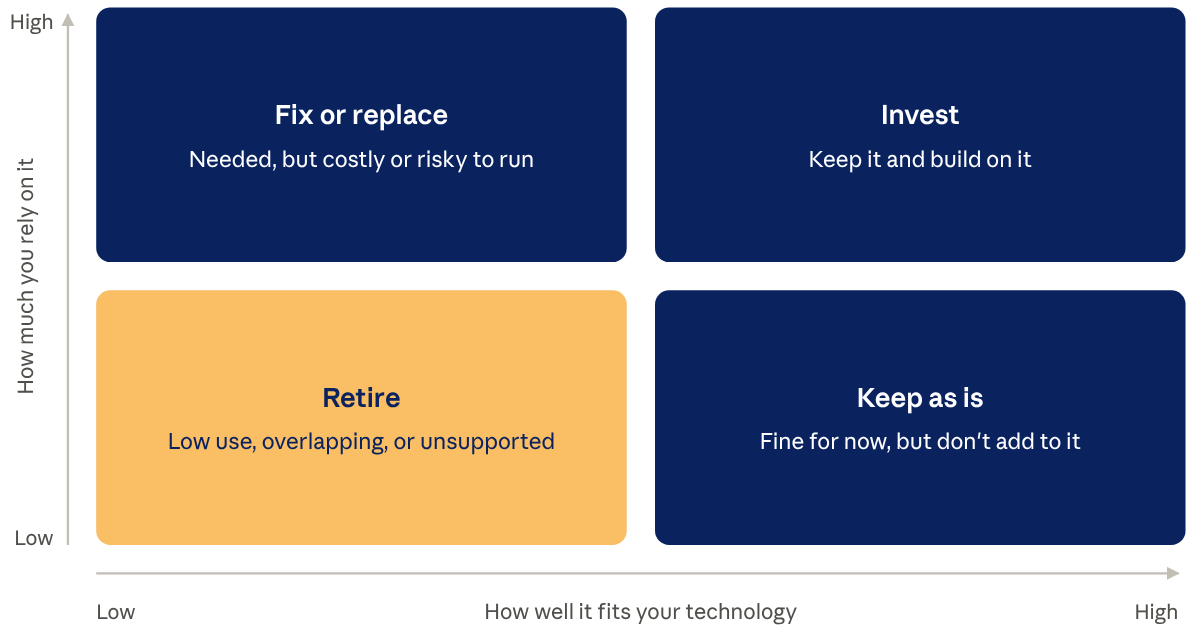
The payoff can be large. Trinity Health reported about \$80 million in savings from retiring legacy applications after its move to Epic, and expects the total to pass \$100 million ([Chief Healthcare Executive](#)).

Sort every app into one of four boxes

Score each app on two questions. How much does the organization rely on it? And how well does it fit your technology: supported, secure, connected, and not duplicated somewhere else? Where it lands tells you what to do next.

Low-use, overlapping apps are the first savings

Sort each app by how much you rely on it and how well it fits your technology



app sorting grid · 4 boxes, 2 questions

Most of the early savings sit in the bottom-left box. Most of the hard work sits in the top-left one.

What a good inventory record holds

Field	Why it matters
Name and what it does	Lets you spot overlap by function, not by vendor name
Business owner and technical owner	Someone has to decide, and someone has to do the work
Full annual cost	License, hosting, support, and interfaces, not just the license
Active users and last login	Usage data settles debates that opinions can't
Connections	Shows what breaks if you turn it off
Data it holds	Drives the archive and retention plan

Field	Why it matters
Contract end date and notice period	Tells you when you can act, and when you must give notice
Support and security status	Unsupported versions are a risk as well as a cost

Results like these are real. Emplify Health in Wisconsin eliminated more than 30 applications over two years, saving \$3 million, plus \$1.3 million in "soft" savings from redirected staff and hardware. Providence cut its application portfolio by more than a third over 12 to 18 months ([Becker's Hospital Review](#)).

Integrate before you add

Every new tool has to fit with what you already run. When it doesn't, people fill the gaps by hand: retyping data, switching between screens, and reconciling records that don't match. That's where much of the administrative drag comes from.

The national picture is improving. TEFCA, the federal framework for sharing health records across networks, passed 1 billion records exchanged in less than a year, according to HHS figures reported in June 2026 ([Becker's Hospital Review](#)). In July 2025, CMS also launched a pledge program for health technology organizations, and networks that pledge to meet its interoperability framework can become CMS-Aligned Networks ([ONC](#)).

Better national connections don't fix integration inside your own walls, though. A few principles help:

- **One source of truth for each kind of data.** Decide which system owns patients, providers, locations, and charges, and have everything else read from it.
- **Use the core platform first.** If your EHR or ERP can do the job well enough, a separate tool adds a login, an interface, and a contract.
- **Standards over custom builds.** Ask vendors for standard APIs, such as HL7 FHIR, instead of one-off interfaces only they can maintain.
- **Know every connection.** Keep a current list of interfaces, who owns each one, and what depends on it. It's your map when something breaks and when you retire an app.
- **Budget for upkeep, not just the build.** Interfaces need monitoring, testing at every upgrade, and someone on call.
- **Make integration a buying requirement.** If a vendor can't show how its tool connects to your systems, it isn't ready.

The new budgeting problem: usage-based AI pricing

AI is changing how software gets priced, and it is catching a lot of health system leaders off guard. Traditional licenses were predictable: a fixed fee per user or per year. Many AI tools now charge by usage, so the bill grows every time someone asks a question or pulls up a record.

It often starts with a low, flat-rate pilot that feels like a no-brainer. Then, at full rollout, pricing moves to usage-based fees, and a cheap experiment becomes a large, hard-to-predict monthly bill. To be fair, that isn't always vendors being sneaky. Sometimes it's simply how scaling AI costs work.

But with margins this thin, end-of-quarter surprises hurt. A few practical guardrails:

- **Read the pilot data closely.** A small, motivated pilot group rarely predicts how a whole organization will use a tool. Model enterprise-wide usage before you sign.
- **Negotiate cost caps up front.** Push for monthly spending ceilings, tiered or bulk pricing, and price protection at renewal before you commit to a usage model.
- **Set clear usage rules.** Turn on high-cost AI features where they save real time or improve care, not everywhere by default.
- **Agree on the bar before you scale.** Decide what return a feature must show, and by when, before rolling it out further.
- **Watch usage monthly.** Track active use by department so a spike shows up in a report, not on the invoice.
- **Govern it together.** Clinical leaders, IT, and finance should own these decisions jointly. Clinical demand and financial predictability both matter.

Predictable budgeting in digital health doesn't happen by accident. It takes structured oversight and real visibility into how tools are used.

Where AI spending is going

Health system AI spending is growing fast. Menlo Ventures estimated that healthcare AI spending reached \$1.4 billion in 2025, nearly triple the year before, with health systems accounting for about \$1 billion of it. Ambient scribes alone made up \$600 million. Buying is speeding up too: health system purchasing cycles shortened from 8.0 to 6.6 months ([Menlo Ventures](#)).

Costs are getting harder to predict. Many AI tools are priced by tokens, the small chunks of text a model reads and writes, so longer notes and bigger models mean bigger bills. Even tools sold as a flat per-provider subscription run on that meter behind the scenes.

Houston Methodist, for one, has limited Microsoft Copilot access to select employees and says it will shut off any AI agent that adds cost without a clear return ([Becker's Hospital Review](#)).

At El Camino Health, the CIO reported price increases of 20% or more across hardware, devices, and software and cloud services, and said the cost of internal large language models is moving to usage-based fees that are hard to forecast. Before accepting any increase, the system now checks usage reports, benchmarks the market, and reruns the cost-benefit math. It also asks vendors to separate AI infrastructure costs from margin ([Becker's Hospital Review](#)).

Contract terms to ask for

Term	What to ask for	Why it matters
Pricing at scale	Fixed or tiered pricing for full rollout, not just the pilot	Pilot pricing often changes at full rollout
Spending cap	A monthly ceiling, with alerts before you reach it	Stops a surprise bill
Usage reporting	Monthly usage by department and user, in a format you can analyze	You can't manage what you can't see
Price protection	Limits on increases at renewal, in writing	Keeps the five-year math honest
Model changes	Notice before the vendor changes the underlying model or its price	A new model can change both cost and results
Performance	Measures the tool must meet, and what happens if it doesn't	Ties the price to value
Exit	Return of your data and help moving off the tool	Keeps you from getting locked in

An illustrative cost model

The numbers below are made up to show the math. They are not real vendor prices.

Say a documentation tool costs \$1 per note under a usage plan. In a 40-clinician pilot where each clinician writes 15 notes a day, that's about \$600 a day. Roll it out to 600 clinicians at the same price and it's about \$9,000 a day, or roughly \$2.3 million a year over

250 working days. If notes get longer or the vendor moves to a pricier model, the per-note cost can rise as well.

None of that makes usage pricing bad. It means the pilot invoice tells you very little about the enterprise invoice, so model it before you sign.

Stop filling the graveyard: buy for real-world conditions

The easiest app to retire is the one you never bought. Discipline at selection keeps the next tool from ending up in the same place.

- **Test against your reality, not the demo.** If a tool only works with perfect data and full staffing, it doesn't belong on your roadmap. Use scripted scenarios built from your own workflows and your messiest cases.
- **Price five years, not one.** Total cost includes implementation, interfaces, internal staff time, support, upgrades, and usage growth.
- **Plan the integration first.** Know how the tool connects to your core platforms, and who maintains those connections, before you sign.
- **Check for overlap.** Ask whether a platform you already own does this today, or will soon. Core vendors add features with every release.
- **Negotiate the exit.** Data return, transition help, and termination terms are far easier to win before signing than after.
- **Name the owner.** Every tool needs an operational owner after go-live and a date to check whether it delivered.

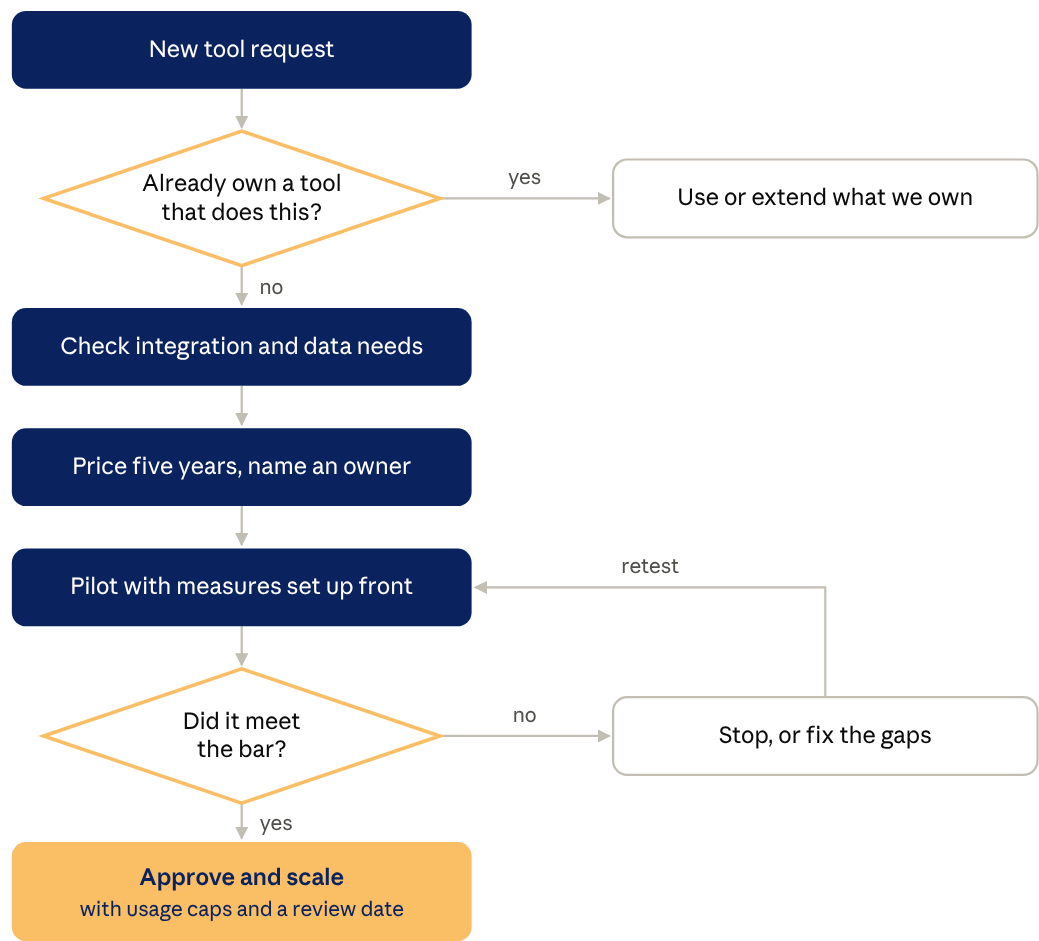
[Our vendor selection approach](#) walks through how we run an objective, requirements-driven selection.

Put every new request through the same gate

A simple, consistent gate keeps the next tool from ending up in the graveyard. It applies to pilots and free trials too, since those are how many tools get in the door.

Every request clears the same gate before anyone signs

Either question can save you a contract



intake gate · 5 steps, 2 decisions

The first question is the cheapest one to answer, and a yes there avoids a new contract altogether.

Metrics to track

A short set of numbers, reported quarterly, shows whether the portfolio is getting healthier or quietly growing back.

Metric	What it tells you	How often
Number of applications, by function	Whether the portfolio is shrinking or growing back	Quarterly

Metric	What it tells you	How often
Share of apps with a named business owner	Whether anyone is accountable	Quarterly
License use rate	How much of what you pay for actually gets used	Monthly for large contracts, quarterly for the rest
Cost per active user	Whether the price still matches the use	At each renewal
Renewals reviewed before the notice date	Whether auto-renewals are under control	Monthly
Number of interfaces	How tangled the stack is, and what each retirement simplifies	Quarterly
AI spending against budget, by tool and department	Whether usage-based costs are staying inside the plan	Monthly
Savings from retirements, and where they went	The return on the cleanup, in dollars	Quarterly, to the executive team

Common pitfalls and how to avoid them

Pitfall	Early warning sign	What to do instead
Treating it as an IT project	Clinical and department leaders aren't in the room	Make decisions jointly with the people who use the tools
Starting with the hardest app	Year one goes to a single complex replacement	Start with low-use, overlapping tools that renew soon
Cutting without a data plan	Retirement stalls when someone asks where the old records went	Plan the archive and retention before announcing a date
Missing the notice window	Contracts auto-renew while the review is still underway	Track notice dates, not just end dates

Pitfall	Early warning sign	What to do instead
Letting the graveyard grow back	New tools get bought outside the intake gate	Send every request, including pilots and free tools, through the same gate
Approving AI on pilot pricing	The first full-rollout invoice lands far above budget	Model enterprise usage and negotiate caps before you scale

What this looks like in practice

An illustrative scenario: a composite built to show how the pieces fit together. It is not a client story, and the numbers are for illustration only.

A regional health system with four hospitals and 40 clinics starts with a simple question from its CFO: how many applications are we paying for?

- **Month 1:** the inventory, pulled from accounts payable, IT, and procurement records, finds about 600 applications, roughly twice what IT expected. Dozens were bought by departments on their own.
- **Month 2:** grouping by function turns up five staff scheduling tools, four secure messaging tools, and three patient survey tools. Usage data shows many licenses haven't been used in months.
- **Month 3:** the steering group picks the first 25 retirement candidates: low use, clear overlap, and renewals coming up within a year. Each gets an owner and a shutdown date tied to its notice window.
- **Months 4 to 12:** retirements happen at renewal. The intake gate turns away two requests that duplicated tools the system already owned. A new AI documentation tool is approved with a monthly cap, usage reporting by department, and a six-month review.

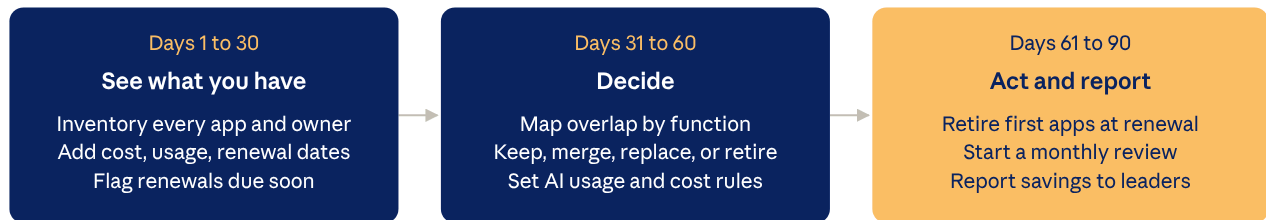
By the end of the year, the CFO gets a quarterly report showing what was retired, what it saved, and where the money went instead.

A 90-day plan to get started

You don't need a multi-year program to start. Ninety days is enough to find the first apps to retire and put guardrails on new AI spending.

A 90-day plan to find your first savings

See what you have, decide, then act



90-day plan · 3 phases

Who to involve

- **An executive sponsor**, often the CFO or COO alongside the CIO, to make the hard calls and clear the path.
- **IT application owners**, who know what each system does and what connects to it.
- **Clinical and department leaders**, who know what people actually use and why.
- **Finance and procurement**, who hold the contracts, costs, and notice dates.
- **Compliance, privacy, and health information management**, who decide what data must be kept and for how long.

What to have in hand at day 90

- A complete inventory with owners, costs, usage, and contract dates
- A first list of retirement candidates, each with a date
- An intake gate for new requests
- Guardrails on AI spending: caps, usage reporting, and a review schedule
- A quarterly report to leadership, with the first savings on it

Questions leaders should ask

- How many applications do we pay for, and who owns each one?
- Which contracts renew in the next 12 months, and when are the notice dates?
- Where do we have more than one tool doing the same job?
- What does each app really cost once you add interfaces, support, and staff time?
- Which AI tools are priced by usage, and what's our worst-case monthly bill?
- What has to be true for a pilot to scale, and who decides?
- How much did retirements save this year, and where did the money go?

Checklist for the first 90 days

- ☐ Name an executive sponsor and a working team
- ☐ Pull every application from finance, IT, and procurement records
- ☐ Add owners, costs, usage, connections, and contract dates
- ☐ Flag every renewal in the next 12 months
- ☐ Group apps by function and mark the overlap
- ☐ Sort each app: invest, fix or replace, keep as is, or retire
- ☐ Pick the first retirement candidates and set dates
- ☐ Set up the intake gate for new requests
- ☐ Put caps and monthly usage reporting on AI tools
- ☐ Send the first savings report to leadership

Start with 40 free hours: Spark's App Rationalization program

If any of this sounds like your portfolio, there's a low-risk way to find out. Spark's App Rationalization program starts with 40 free hours of senior consultant review.

SPARK APP RATIONALIZATION PROGRAM

40

free hours of application rationalization

Delivered by senior healthcare consultants

Less than 10 hours of your team's time · About six weeks from kickoff to executive readout

Spark App Rationalization program · the starting offer

There's no cost, no follow-up commitment, and no procurement cycle. If the work is useful, we talk about a second phase. If not, you keep the deliverable.

How the 40 hours work

Four steps, and less than 10 hours of your team's time:

Step	What happens	Your team's time
Discover	We pull your app inventory, contracts, and license data	About 2 hours
Map	We group apps by function, owner, spend, and usage	About 3 hours
Rationalize	We find redundancies and consolidation opportunities	About 2 hours
Recommend	You get a prioritized list of opportunities with savings estimates	A 1-hour readout

What the six weeks look like

1. **Week 0, kickoff:** a kickoff call, and we send a data request list.
2. **Weeks 1 and 2, discovery:** inventory pull and stakeholder interviews.
3. **Weeks 3 and 4, mapping:** we group apps, find redundancies, and build the rationalization model.
4. **Week 5, draft review:** a working session with your IT leadership to check the findings.
5. **Week 6, executive readout:** we hand off the final deliverable and talk through what a second phase could look like.

Who does the work

Every hour comes from a senior consultant, not junior analysts learning on your time. Our team has worked as healthcare operators, so we know EHRs, ERPs, and the third-party apps around them. And because we don't resell software, our recommendations aren't shaped by any vendor relationship.

What it can turn up

At one Philadelphia-area health system (name withheld at the client's request), Spark assessed 330 third-party applications and found more than 170 that were redundant or unused. The result: a 52% smaller portfolio, seven-figure annual savings identified in licensing and support contracts alone, three vendor consolidations the IT team had long wanted, and the business case the CIO used to fund a full rationalization program.

Curious what's sitting in your graveyard? The first step is a short scoping call. [Start a conversation.](#)

Sources

- [The hospital finance divide: 15 things to know](#), Becker's Hospital Review, Oct. 19, 2026, citing Kaufman Hall's National Hospital Flash Report (August 2026 data from 1,300 hospitals)
- [How much is wasted on SaaS spend](#), Zylo, updated March 19, 2026, citing Zylo's 2026 SaaS Management Index
- [76% of CIOs call application rationalization 'critical': CHIME survey](#), Becker's Hospital Review, Feb. 11, 2026
- [Health systems save millions by eliminating IT applications](#), Becker's Hospital Review, Feb. 25, 2026
- [Trinity Health talks about saving nearly \\$80M at ViVE 2026](#), Chief Healthcare Executive, Feb. 12, 2026
- [What's new with TECCA in 2026: 4 updates](#), Becker's Hospital Review, updated July 1, 2026
- [The Tide and the Speedboats: TECCA and CMS-Aligned Networks](#), ONC blog, Dec. 16, 2025
- [2025: The State of AI in Healthcare](#), Menlo Ventures, Oct. 21, 2025
- [The hidden cost of AI tokens](#), Becker's Hospital Review, Aug. 3, 2026
- [AI chip demand is driving up El Camino Health's tech bills](#), Becker's Hospital Review, July 24, 2026
- [Spark Healthcare Advisory on LinkedIn](#), posts on software roadmaps, app rationalization, and AI pricing, Aug. to Oct. 2026