

Choosing the right interoperability partner

Critical questions to ask before you commit

Why interoperability decisions matter

The integration partner you choose shapes your speed to market, your operational burden, and your long-term cost structure. Most integration challenges don't appear at go-live — they surface months or years later as you scale. Asking the right questions now can prevent hidden staffing needs, unexpected services costs, and ongoing maintenance risk. Use this guide to evaluate any partner consistently and confidently.

Ownership & resourcing

Questions to ask:

- Who maintains integrations after go-live: your team or ours?
- How many internal engineers or integration specialists will we need in year one? In year three?
- What specific technical skillsets must our team have? (e.g., HL7v2, FHIR, network engineering, API development)
- Who monitors connections 24/7/365 and resolves issues when they arise?
- If we don't have certified project managers or developers with healthcare integration experience on staff, do you offer those resources — and at what cost?
- Can non-technical staff monitor, troubleshoot, or manage integrations?

Strong signals:

- ✓ Clear, defined ownership model post-go-live
- ✓ Minimal required internal FTEs for ongoing management
- ✓ Proactive monitoring and issue detection, not reactive troubleshooting
- ✓ Vendor acts as an extension of your team, not just a tool
- ✓ Ability to manage and monitor integrations via natural language

Why this matters: Integration projects rarely fail at launch. They fail in long-term maintenance and operational ownership. The staffing model your vendor requires is often the biggest hidden cost in your total budget.

Scalability & architecture

Questions to ask:

- Does each new connection require custom development from our team, or can configurations be reused?
- How quickly can new endpoints or interfaces be added if our scope changes?
- How does your platform help us accelerate our 10th, 20th, and 50th connection — not just our first?
- What's the process for scaling across dozens of systems without re-building each time?
- Does your platform expose integration data to AI agents and tools — or only to human users through a UI?

Data & technical reality

Questions to ask:

- How is incoming data normalized across different systems, formats, and standards?
- What happens when we encounter non-standard fields or custom segments (e.g., Z-segments in HL7)?
- Can our engineers access full API documentation and a sandbox environment before we sign a contract?
- Who owns data mapping when an upstream system changes — and what does that cost?
- How does your platform handle modern standards like FHIR R4/R5 — natively, or through bolt-on modules?
- Do you support discrete, bi-directional write-back into source systems, or is it limited to static documents (PDFs/CDAs) and flat text?

Strong signals:

- ✓ Reusable configurations and pre-built mappings across connections
- ✓ One-to-many architecture
- ✓ Predictable expansion timelines regardless of volume
- ✓ No need to rebuild integrations as your partner network grows
- ✓ Native support for MCP (Model Context Protocol)

Why This Matters: What works for one integration must scale without multiplying your team's complexity or cost. A solution that requires custom development for every connection will become a bottleneck as you grow.

Strong signals:

- ✓ Automated normalization across legacy and modern standards
- ✓ Native standards support, no bolt-ons required
- ✓ Transparent, publicly available API documentation
- ✓ Clear ownership of mapping changes post-launch with predictable cost

Why This Matters: Moving data is not the same as delivering reliable, usable data. Normalization, mapping ownership, and standards support determine whether your integration holds up in production — or creates ongoing technical debt.

Long-term cost & operational impact

Questions to ask:

- Does this proposal represent the total expected cost — direct and indirect? What's not included?
- What will version upgrades cost, and how disruptive will they be?
- What additional cloud or infrastructure costs should we expect?
- What is the cost and process for configuration changes or new connections six months after go-live?
- Who writes the project plan, tracks milestones, and manages escalations with third-party partners?
- Will we need to file a support ticket for every change — or can our team handle it directly?

Strong signals:

- ✓ Predictable, transparent pricing model
- ✓ Continuous updates instead of disruptive major upgrades
- ✓ No surprise infrastructure or cloud processing costs
- ✓ Post-launch changes covered without a new SOW

Why This Matters: Implementation cost is only part of the equation. Year-2 and Year-3 operational burden — hidden services fees, upgrade projects, and internal staffing needs — determine your real ROI. Make sure you're comparing total cost of ownership, not just contract price.

Before you decide

Before selecting a partner, make sure you can confidently answer.

- ☐ *How many internal FTEs will we need, long term, not just at launch?*
- ☐ *What will this cost in year three, not just year one?*
- ☐ *How quickly can we expand without rebuilding?*
- ☐ *Who owns uptime, monitoring, and escalations?*
- ☐ *What happens when standards evolve or a partner changes their system?*

Connect once. Exchange anywhere.