Omilia

Conversational Intelligence

Build vs. Buy in the Age of AI Coding

The Sustainability Calculus for Enterprise Conversational AI

May 25, 2026

Authored by,

Dimitris Vassos,
Founder & CEO

Executive Summary

Disclosure. I am the founder and CEO of Omilia, a global leader in conversational AI platforms, and I have worked on AI for contact-center and customer-service automation for more than twenty years. The analysis below draws on that vantage and on the independent industry research cited throughout.

The build-vs-buy decision for enterprise conversational AI has crossed a structural threshold, because AI-assisted coding has collapsed the cost of producing software while leaving the cost of owning it intact. The reflexive inference, that cheaper creation justifies more in-house building, draws the wrong conclusion from the right observation. The decision criterion that mattered for three decades, framed by cost, control, and customization, no longer carries the analytical weight that the current generation of enterprise leaders attribute to it. According to Philipp Schloter, writing in the Forbes Business Council on 17 November 2025, the question has shifted from whether the organization can build a capability to whether it can sustain it as the world changes.

Three Strategic Planning Assumptions published by Gartner between September 2025 and April 2026 converge on the same conclusion as Schloter's framing, from three different analyst teams working on three different research streams:

- "Most agents built before 2028 will need replatforming or rebuilding by 2030." (Source: Gartner, "How CIOs Build, Buy and Partner to Scale AI Capabilities," Stewart Buchanan, 28 April 2026.)
- "By 2027, 70% of organizations that decide to build their own RAG will see their total cost of ownership over three years surpass their initial budget by more than twofold." (Source: Gartner, "Decide Between Build or Buy Solutions for RAG," Tong Zhang, Haritha Khandabattu, Xingyu Gu, Darin Stewart, 11 September 2025.) RAG denotes retrieval-augmented generation.
- "By 2027, over 40% of agentic AI projects will be canceled." (Source: Gartner, "Top Actions to Drive Success in Building Agentic AI Solutions," Aaron Harrison, Haritha Khandabattu, 9 April 2026.) The cancellation drivers Gartner names are escalating costs, unclear business value, and inadequate risk controls.

This paper is written for chief technology officers and senior engineering leaders evaluating conversational AI in enterprise customer service and contact-center settings, the domain in which the build, buy, and blend tension is sharpest. The argument proceeds in six moves: most "built in-house" conversational AI is an orchestration layer on a third-party inference API; the 10/90 lifecycle problem has been refreshed with quantified analyst data; AI coding compressed build time symmetrically, so competitive distance now sits in post-launch evolution where the platform vendor's market exposure is structurally higher; the same cross-customer exposure produces industry-specific data at a scale no single enterprise can match; the canonical Gartner answer is blend rather than binary; and cost reduction is the first wave of customer-service AI value, not the whole game (Vicchi, G00845891).

The synthesis is that the buyer should build what is genuinely differentiating and sustainable for five years, buy what is essential infrastructure, and blend the two with disciplined architecture in between. The criterion that decides the call is no longer buildability; it is the organization's exposure to the market signals that drive post-launch evolution.

1. The Build-vs-Buy Calculus Has Shifted: Three Strategic Planning Assumptions

For three decades the build-vs-buy decision rested on a familiar checklist: cost, control, and customization. Schloter, in the Forbes Business Council on 17 November 2025, argues the checklist no longer tells the full story for enterprises operating in cloud, AI, and continuous-evolution conditions (Schloter, Forbes Business Council, 17 Nov 2025). Cloud platforms shift continuously, AI shifts faster, and regulatory and data standards do not hold still long enough for a build-once-deploy-forever program to remain compliant. The question has migrated from whether the organization can build a capability to whether it can sustain it as the world changes, which is a question about operating model rather than technology procurement.

That framing is not provincial to a finance-transformation columnist. It maps cleanly onto the Gartner analyst record, where the same observation appears in three different research streams: the sustainability cost of agentic AI (Buchanan, G00853271), the cost-of-ownership overrun on in-house RAG (Zhang et al., G00837989), and the cancellation rate for agentic AI projects before 2027 (Harrison and Khandabattu, G00846798). Different evidence, similar conclusion. Buchanan's note frames the underlying point most directly: AI sourcing belongs in the operating-model category rather than the technology-procurement category, and capabilities the enterprise cannot govern, orchestrate, or evolve tend to accumulate operational obligations on a steeper curve than the value they return.

The mechanism, in plain terms: buildability has ceased to function as a credential. AI-assisted coding now produces a serviceable first version in days, so the first version is no longer where competitive distance is established. What carries the weight buildability used to carry is whether the organization can sustain the system through five years of model deprecations, regulatory revisions, channel migrations, and personnel turnover.

For the CTO, the practical implication is precise. The system worth having is the one that runs for five years through those transitions. **The system that ships fast and ages faster is a liability with a delayed bill**, and the bill is denominated in replatforming effort that does not appear in the original business case.

The Three Strategic Planning Assumptions

The empirical spine of the build-side risk argument rests on three Strategic Planning Assumptions, each independently sourced from Gartner research notes published between September 2025 and April 2026, and each authored by a distinct analyst team. Read together, they describe the same durability problem from three different angles: replatforming horizon, cost-of-ownership overrun, and project survival rate.

Buchanan, in "How CIOs Build, Buy and Partner to Scale AI Capabilities" (G00853271, 28 April 2026), states the replatforming assumption: "Most agents built before 2028 will need replatforming or rebuilding by 2030." In practical terms this is a two-year shelf life for AI agents built in the current generation, after which the organization faces a forced re-engineering event whose cost is rarely captured in the original build's business case. The same note frames the underlying mechanism as the operating-model exposure described above,

with capabilities that the enterprise cannot govern, orchestrate, or evolve accumulating operational liabilities on a steeper curve than the value they return.

The two-year shelf life is the modal outcome Gartner predicts, not a long-tail risk. For the CTO who has already approved an in-house agent build, the replatforming event is already on the calendar even if no one has yet added it, and the business case that did not budget for it was built against the wrong cost line.

Zhang, Khandabattu, Gu, and Stewart, in "Decide Between Build or Buy Solutions for RAG" (G00837989, 11 September 2025), state the cost-of-ownership assumption: "By 2027, 70% of organizations that decide to build their own RAG will see their total cost of ownership over three years surpass their initial budget by more than twofold." The cost drivers named are not exotic engineering specialisms but the recurring work of any production retrieval pipeline: chunking and embedding strategy selection, hybrid search integration, ranking refinement, access control on sensitive data, and dynamic embedding for enterprise-grade RAG. Each of those is a discipline that requires sustained engineering attention. Taken together they describe an ongoing engineering organization, not a one-time delivery.

Harrison and Khandabattu, in "Top Actions to Drive Success in Building Agentic AI Solutions" (G00846798, 9 April 2026), state the cancellation assumption: "By 2027, over 40% of agentic AI projects will be canceled." The cancellation drivers Gartner names are escalating costs, unclear business value, and inadequate risk controls, each of which surfaces during the years after the initial build rather than during the prototype. Gartner's prescription is not "do not build," because the document is written for software engineering leaders who will continue to build. The prescription is that the preconditions for a successful build are steeper than commonly understood, with data readiness, RAG as a platform capability supported by data contracts and hybrid retrieval, and an agent-aware development lifecycle distinct from traditional software development lifecycle (SDLC) being the named requirements.

Three SPAs, three analyst teams, three lenses, same direction: the durability of in-house AI builds is where the economics break, not in the prototype but in the years that follow. The buyer contemplating a build should treat these assumptions as the baseline against which the proposed five-year case must be tested, with the burden of proof on the build advocate.

2. The Inference-API Boundary, or the “Built In-House” Illusion

Most “built in-house” conversational AI is not built in-house in any architecturally meaningful sense, because it is an orchestration layer running on top of a third-party inference application programming interface (API). The conversational stack is not one component; it is a sequence of provider calls strung together, and naming the sequence explicitly is the precondition for any honest discussion of what “in-house” means in this context.

A production conversational AI system in the contact-center context typically runs the following stack on every customer call:

- **Speech to Text (STT)** — the audio of every utterance is transcribed to text, on every turn, with a latency budget measured in tens of milliseconds and an accuracy requirement that varies by domain, accent, and channel quality.
- **Intent classification and NLU** — the transcribed utterance is mapped to an intent and slot structure, with confidence scoring that drives downstream policy decisions and that itself becomes a separate provider call when wrapped on a foundation-model API.
- **Zero shot Intentless NLU** — the model handles inputs it has never seen during training, without classifying utterances into named intents. Instead of a rigid intent catalog, it understands meaning directly from language — processing novel, open-ended inputs and deriving context without requiring a predefined intent framework.
- **Dialog state management** — the conversation is tracked across multi-turn sessions, with state held in a memory layer that the bot consults on every subsequent turn, including resolved entities, pending confirmations, and escalation flags.
- **Natural language generation (NLG)** — the bot’s response is composed in language conditioned on the dialog state, the brand voice, the regulatory constraints applicable to the utterance, and the channel surface.
- **Text-to-speech (TTS)** — the generated response is rendered to audio that matches the channel’s codec and latency budget, with prosody and persona settings that the buyer expects to control.
- **Voice biometrics, Spoof Detection, and Antifraud** — authentication and fraud-pattern detection run as parallel inference paths on the audio, with the biometric component trained on years of voiceprint data and the antifraud component trained on fraud-pattern data the buyer does not have at scale.
- **Agent-assist signals** — at handoff, the bot’s context, transcript, and recommended next actions are streamed to the human agent, with the signal quality determining whether the handoff reduces handle time or increases it.

Each of these components, when wrapped on a foundation-model API, makes a separate provider call. The buyer’s system is the orchestration and integration layer; the provider performs the conversational work.

The architectural implications accumulate across six independent dimensions:

- **Sovereignty** — customer conversation data crosses a third-party boundary at runtime, on every call, which is a sovereignty question rather than a configuration question.
- **Behavior Stability** — model behavior can change across provider version updates with limited notice, which means the contact-center quality assurance baseline is set by the provider and not the buyer.
- **Cost Predictability** — pricing is set unilaterally by the provider, which transforms a forecastable cost line into a unilaterally-revisable one.
- **Migration Exposure** — the provider can deprecate a model version on its own timeline, which exposes the buyer to forced migration events the provider does not coordinate.
- **Compliance Inheritance** — the provider's compliance posture becomes, in practice, the enterprise's compliance exposure, because every audit boundary the enterprise crosses inherits the provider's controls.
- **Contract Leverage** — the contract surface is an API contract the buyer did not write and cannot renegotiate, which removes the leverage that enterprise procurement typically exercises in long-running vendor relationships.

Rigon, Elliot, Tung, and Bhatia (G00839531, 19 September 2025) frame the cost dimension of this directly. Their note observes that buyers routinely understate what it takes to keep a conversational AI system running across construction, upkeep, orchestration, and scale, and that the understatement pushes program owners into short-horizon decisions whose downstream cost is paid in cumulative engineering debt and run-rate growth. That is the analyst register for what happens when an enterprise treats a foundation-model API as if it were owned infrastructure, because the misestimate is not principally an estimation error but a category error about what is being acquired.

The voice channel adds constraints that text-only chatbot patterns do not surface. End-to-end latency has to sit under roughly 200 milliseconds to feel natural, a budget split across ASR, NLU, dialog, NLG, and TTS every turn, with provider calls contributing their round-trip latency. Multilingual coverage is a years-of-data investment per language, not a configuration toggle. Voice biometrics requires multi-year training data per voiceprint population, and antifraud requires fraud-pattern data the buyer does not have at scale. These are the standard load of any production voice deployment, and they are where wrapper architectures fail.

The diagnostic question reduces to a single sentence. When the system handles a customer call, whose servers does the audio and the transcript touch at inference time? If the answer includes a foundation-model provider's infrastructure, the system is leased capability with a self-managed wrapper, and the operating-model exposures above apply.

3. The 10/90 Lifecycle Problem, Refreshed

In enterprise environments, the initial build of a non-trivial system typically represents 10 to 20 percent of total lifecycle effort, while the remaining 80 to 90 percent goes to maintenance, security patching, integration upkeep, compliance updates, governance, and the organizational cost of knowledge concentrated in a small number of engineers. AI coding tools have compressed the visible 10, often dramatically, while leaving the invisible 90 substantially unchanged. The first-order consequence is that build-side proposals systematically understate true cost, because the proposal is written against the cost line that AI tools have changed rather than the cost line that determines the five-year outcome.

The analyst record refreshes the 10/90 proportion with quantified data. The three SPAs detailed in §1 (Zhang on RAG TCO, Harrison on cancellation, Buchanan on replatforming) are modal outcomes Gartner predicts across the population, not edge cases.

The compliance load is heavier than the general enterprise software case. PCI-DSS scope applies to card numbers on voice; HIPAA to PHI utterances; GDPR to EU residency, retention, and right-to-be-forgotten; SOC 2 Type II to enterprise procurement; ISO 27001 and CCPA in the markets where they apply. Each is a multi-quarter project with annual recertification, and the in-house team that did not budget for any of them will absorb them as unplanned engineering work.

Schloter's total-cost-of-ownership decomposition (Forbes Business Council, 17 Nov 2025) names the hidden line items that operating-model accounting tends to omit:

- **Employee Time** — minutes per task multiplied across the workforce, with the resulting line item often exceeding the platform subscription it was meant to replace.
- **Training and Support** — retraining whenever the user interface or the underlying model changes, with the cadence of change rising as the AI stack evolves.
- **Maintenance** — security patches, regression testing, version updates, with the patch cadence set by upstream providers rather than by the buyer.
- **Compliance Exposure** — audit findings, exceptions, certification work, on top of the certification stack named above.
- **Switching Costs** — migration effort, contract renegotiation, data-export friction, which determine the realism of the exit option that the lock-in critique assumes the buyer holds.

The architectural lesson is that the build budget shows the visible cost while the ownership budget determines the outcome. A five-year case that fails to model the ownership budget rigorously is not a case; it is a forecast of the prototype that omits the system.

4. The Innovation-Velocity Asymmetry

The instinct that AI-assisted coding tilts the build-vs-buy decision toward build draws the correct observation from the wrong premise. The observation is that AI coding has collapsed the cost of producing the first version. The premise that does not survive examination is that this collapse benefits the in-house team disproportionately. It does not. The same tooling, the same productivity multipliers, and the same compression of the build phase are available to specialized platform vendors, who use them on the same code paths that the in-house team is building. The asymmetry is not in build velocity; it is in what happens after the first version ships.

A specialized platform vendor sees emerging requirements across hundreds of enterprise deployments, across markets, regulatory regimes, and channel surfaces. The signal density is high, the feedback is continuous, and the requirements that surface in one deployment frequently anticipate those that will surface in the next ten. The in-house team sees the requirements of one enterprise, filtered through one business's internal vocabulary, with the feedback loop running in a closed circuit.

The consequence for roadmap evolution is structural rather than incidental. The platform's roadmap is shaped by market signals the enterprise cannot generate from within its own walls, because the signals that drive the roadmap come from outside the enterprise. New channel surfaces, new compliance regimes, new fraud patterns, new language demands, new latency expectations, new agent-handoff models: each of these enters the platform's roadmap through deployments the in-house team does not see. The in-house artifact is born scope-bound, with its requirements defined by the originating business's current understanding of its problem. The platform artifact is born market-exposed, with its requirements continuously refreshed by the deployments the platform serves.

Over a five-year horizon, the in-house build does not just fall behind on raw capability. It falls into a constrained scope the originating enterprise no longer sees as constrained, because the team's vocabulary for the problem was set at the original build and has not been refreshed by external signals since. The gap is invisible from inside the building, visible only when a competitor on a platform demonstrates a capability the in-house team had not modeled as a requirement, at which point the cost of catching up is the cost of replatforming on top of the original build. Rigon et al. (G00839531), referenced in §2 on the chronic understatement of ongoing CAI effort, name the same dynamic: the work that gets understated is principally the work of staying market-relevant in a domain whose requirements the in-house team does not directly see.

The CTO test, in light of this asymmetry, is not whether the in-house team can deliver the first version. AI-assisted coding has made that question almost trivial, and the analyst record's repeated emphasis on the cancellation rate (Harrison and Khandabattu, G00846798) reflects the failure mode that follows from passing only that test. The test is whether the in-house team has the market exposure to drive a roadmap that keeps the system competitive after launch. In most enterprises, that exposure does not exist, because the engineering organization is not in the market; it is in the business. Buchanan (G00853271) names the resulting position the buyer-as-orchestrator, with partner roadmap velocity the load-bearing reason for relying on partners to fill capability gaps.

The implication for the procurement conversation is that velocity, narrowly defined as the rate at which the first version ships, should be removed from the decision criteria. Both build and buy ship the first version on

similar timelines, with AI-assisted coding having collapsed that timeline symmetrically. What remains is the rate of post-launch evolution, and on that criterion the platform's market exposure is the structural advantage that the in-house team cannot replicate by staffing alone.

5. The Industry-Data Asymmetry

The asymmetry described in §4 establishes the post-launch evolution gap. The same cross-customer exposure produces a second asymmetry, parallel in structure and equally consequential: industry-specific data. A platform vendor serving multiple enterprises in banking, insurance, healthcare, or utilities sees, across those deployments, the dialog patterns, intent taxonomies, regulatory language, fraud patterns, customer-language variation, and escalation models that define each industry's conversational surface. The in-house team sees the conversational surface of one enterprise, which is a single sample drawn from a distribution whose shape only the platform vendor can characterize.

What "industry-specific data" actually means in a conversational-AI context deserves explicit accounting, because the abstraction hides the substance. A platform serving twenty banks observes signals across categories that no single bank can assemble at meaningful volume:

- **Cross-bank intent taxonomies** — the catalog of customer goals as they actually surface in production, with the relative frequency of each intent and the seasonal, demographic, and channel variations that shape that frequency.
- **Dialog patterns for high-value journeys** — account-balance, dispute, fraud-claim, card-replacement, and loan inquiry flows across customer cohorts of different ages, primary languages, and channel preferences, with the resolution paths that work and the ones that produce escalation.
- **Regulatory-language variation across jurisdictions** — the same regulatory concept (KYC, AML, GDPR consent, US fair-lending disclosure) producing different conversational surfaces in EU, UK, US, and APAC deployments, with the platform vendor seeing the cross-jurisdictional spread that any single bank's deployment cannot.
- **Fraud-pattern coverage at industry scale** — pattern volume and recency that no single bank can match, with the platform seeing patterns emerging in one customer's deployment before they reach the next, which transforms fraud detection from a backward-looking analytic to a forward-looking signal.
- **Accent, dialect, and language coverage at population scale** — STT and NLU training data spanning the industry's full customer geography, with rare-accent and minority-language coverage that any single enterprise's deployment cannot generate.
- **Agent-handoff patterns and dialog-state signals** — the predictive features that distinguish a successful handoff from a failed one, observed across thousands of agent interactions per deployment and millions across the customer base.

Each of those categories is a data asset the platform vendor accumulates as a structural byproduct of serving the industry, and each is an asset the in-house team cannot replicate from its own deployment alone.

Data accumulation is a precondition, not a finished capability. The operational asymmetry follows from what the platform vendor does with the data: distill large foundation models into smaller industry-tuned variants, apply parameter-efficient fine-tuning (LoRA and equivalent) on industry corpora, evaluate against industry-specific test sets that no general benchmark covers, and retrain on a continuous cadence as the data

refreshes. The resulting models are smaller, faster, cheaper to serve, and more accurate on the industry's task distribution than a generalist API can be, because the generalist API is optimized for a distribution that includes the industry's traffic as a small fraction of its training mix. The buyer of the platform inherits the industry-tuned model on day one. The in-house team inherits a generalist API, with the option to spend years closing the gap.

The rebuttal that an enterprise can fine-tune the models itself is the predictable counter, and it does not survive contact with the operational requirements. To produce an industry-tuned model that the enterprise owns and controls, four programs must be undertaken in parallel:

- **Adopt open-source foundation models** — because closed proprietary APIs either do not permit fine-tuning at depth or restrict it to managed-service arrangements that do not produce a self-owned model, the enterprise must shift to the open-source class (Llama, Qwen, Mistral, or equivalent), with the licensing, model selection, and version management that follows.
- **Bring inference in-house or onto an open-source-friendly stack** — vLLM, TGI, or equivalent serving infrastructure must be operated, with the GPU procurement, batching, KV-cache management, autoscaling, and reliability engineering that production inference at enterprise scale requires.
- **Build a deep ML engineering and MLOps team** — dataset curation, fine-tuning, evaluation, regression testing, drift detection, model registry management, A/B testing, and the operational cadence of continuous retraining each require specialist staffing and tooling, with the resulting team representing a permanent line item rather than a project budget.
- **Acquire and curate industry-grade training data** — which the single enterprise does not have at the volume that the platform vendor accumulates across its customer base, with the data acquisition problem unsolvable inside one enterprise's deployment footprint.

The enterprise that takes all four on has become a platform vendor with one customer: the worst position on every column of the framework table in §9. The full operational complexity of self-hosted inference (Chandrasekaran, Ramos, and Barot, G00809217, 2 June 2025) compounds the build-side RAG TCO that Zhang et al. (G00837989) already place at more than twice the initial budget.

§4 framed the CTO test as market exposure for roadmap evolution; §5 frames it as data exposure for model quality. Both tests are answered the same way for the substantial majority of enterprise buyers, and that convergence is the structural reason Gartner's blend-or-buy recommendation has held across analyst teams.

6. The Conversational AI Build-vs-Buy Spectrum

Gartner's canonical model for sourcing conversational AI, set out by Rigon, Elliot, Tung, and Bhatia in "Select the Most Fitting Approach to Enable Conversational AI" (G00839531, 19 September 2025), plots four solution archetypes on a build-vs-buy spectrum, each with a distinct risk-and-effort profile. The four archetypes, read from the buy end of the spectrum to the build end, are summarized below.

- **GenAI-native applications (most "buy")** — stand-alone generative-AI-first SaaS products with dedicated user interfaces, typically chatbot-only consumer-facing assistants. Rapid deployment, limited use-case scope.
- **Targeted CAI extensions inside enterprise applications** — conversational-AI add-ons within ITSM, contact-center, and similar enterprise application stacks. Suitable for medium-scope use cases with high TRiSM readiness in mature extensions.
- **Dedicated conversational AI platforms (CAIPs)** — standalone software products built for conversational applications across channels (voice, chat, messaging, email), using composite AI. Rich feature set, suitable for high-risk and high-exposure use cases, with a steeper learning curve.
- **Custom AI solution development environments (most "build")** — low-level tooling for bespoke AI, including LLMs and RAG. In conversational AI this is the LLM-plus-RAG-plus-orchestration build, with the team assembling ASR, NLU, dialog management, NLG, and TTS components on top of foundation-model APIs and open-source libraries. Limitless customization, with high complexity that often confines the work to limited-scope pilots.

Andrews and Hare, in "How to Decide Whether to Build, Buy or Blend Your AI Projects" (G00825235, 9 April 2025), layer a use-case classification over the four-archetype spectrum, distinguishing:

- **Defend** — AI deployed to maintain competitive parity, with the success criterion being that the organization keeps pace with the market baseline.
- **Extend** — AI deployed to transform an existing process or team for competitive differentiation, with the success criterion being measurable improvement on a metric the organization owns.
- **Upend** — AI deployed to disrupt with new value propositions, products, or markets, with the success criterion being the creation of a new line of business.

For defend-category use cases, Gartner states the recommendation plainly: "Building AI for a defend-based project is unwise as the effort is so much greater than acquiring it from a vendor who focuses on the general need." (Source: Gartner, "How to Decide Whether to Build, Buy or Blend Your AI Projects," Whit Andrews, Jim Hare, 9 April 2025.) Build is the appropriate choice when the capability is extend or upend, with the precondition that the engineering organization can sustain the build for the five-year horizon described in §1. For everything else, including the majority of enterprise customer-experience automation, blending is the rational default.

Most enterprises in customer service classify their CX automation as extend or upend when the honest classification is defend. The bank's competitive position is in its financial products and service relationships,

not its conversational AI. The utility's position is in service reliability, not IVR replacement. The insurer's position is in underwriting and claims-resolution policy, not the claims-handling bot. The conversational AI is the channel through which the differentiating capability is delivered, not the differentiating capability itself.

7. Cost Reduction is Wave One, Not the Whole Game

Vicchi, in "Look Beyond Cost Savings for Lasting AI Benefits in Customer Service" (G00845891, 2 March 2026), describes a three-wave model for AI value creation in customer service, with each wave representing a distinct relationship between automation and competitive outcome:

- **Wave one is Cost Reduction** — automate interaction volume, reduce per-contact cost. This is the phase that most organizations are in currently, with the dashboard centered on containment rate, average handle time, and per-contact dollar cost.
- **Wave two is Service-Quality upgrade** — use AI to lift the customer experience itself, so that customer experience becomes a competitive lever rather than an expense line, with the dashboard adding NPS, CSAT, and resolution quality alongside the wave-one cost metrics.
- **Wave three is Transformation** — AI reshapes customer engagement, product strategy, and growth, with the consequence that the contact center transitions from a cost center to a strategic surface, and the dashboard adds revenue contribution, retention lift, and cross-sell conversion.

Vicchi's central caution is direct: "Organizations that approach each wave sequentially will see a slump in AI implementation value as they shift." The mechanism, in Vicchi's note, is that wave-one cost-driven initiatives lose relevance over time and the retooling required to shift into wave two is "more expensive and less efficient" than a CX-aware approach designed for waves two and three from the start.

The wave-one architecture that wins the first business case is rarely the architecture that survives wave two. Wave one optimizes for containment, with a flat dialog and a narrow intent taxonomy that maximizes deflection without regard to perceived quality. Wave two requires multi-turn reasoning, brand-voice fidelity, escalation policies that respect the customer's stated emotion, and an agent-handoff signal the human agent can act on. The wave-one stack becomes the wave-two stack by rebuild, not by configuration, and for the in-house team that won approval on the wave-one cost case, the rebuild is the second business case after the first.

Three additional cautions deserve direct treatment. Cost reduction must not be the sole focus: advantages erode quickly when new infrastructure costs land. The myth of agentless service must be avoided: automation reconfigures the workforce rather than eliminating it. Sustained technology expenditure must be planned for: AI platforms and supporting infrastructure may outpace the savings automation generates.

Gartner names two metrics for tracking CX-led AI value beyond cost reduction:

- **The AI-Driven Agent Impact Ratio** — defined in Vicchi's note as (redeployed agents / displaced agents) × 100, which corrects the wave-one dashboard's silent assumption that displacement is the only metric.
- **The CX Impact Ratio** — defined in Vicchi's note as ((post-AI CX score − pre-AI CX score) / pre-AI CX score) × 100, which corrects the wave-one dashboard's silent assumption that customer experience is unchanged or improved by default.

Both metrics are explicit corrections to first-year-savings dashboards, because both refuse to count cost reduction as the only output of the AI program. The five-year horizon is the relevant one: first-year savings will not survive contact with wave two, and a build optimized for wave-one cost has a short window before the world it was built for stops being the world it operates in.

8. Risks of Buying: Lock-In, Roadmap, Acquisition

A vendor-neutral treatment of the build-vs-buy question requires that the buy-side risks be addressed with the same rigor as the build-side risks, because the analyst record itself does not endorse buy as an unconditional default. The blended recommendation across Gartner's recent research is not "always buy"; it is "almost never pure build, and rarely pure buy without disciplined diligence." Three risk categories deserve direct treatment in the procurement process: lock-in, feature gaps and roadmap responsiveness, and acquisition exposure.

Lock-in. Rigon et al. (G00839531) flag that dedicated CAIPs typically ship as self-contained products, which means they integrate less naturally with the enterprise application estate the buyer already owns than an extension to one of those applications would. That integration gap is a real and recurring friction point in procurement. The recommended counter-measures are well-defined in the analyst record. Chandrasekaran, Ramos, and Barot (G00809217, 2 June 2025) recommend the use of AI gateways and similar abstraction layers so the buyer can swap providers without redesigning the application stack, the adoption of open-source frameworks so the deployment is not bound to any single cloud provider's proprietary surface, and the institution of FinOps practices so that cost can be monitored and optimized as the deployment scales. The lock-in critique is legitimate, while the mitigation set is also clearly specified, which means the buyer's leverage in negotiation lies in requiring the mitigations to be designed into the architecture from contract signature. A procurement that signs without specifying data-export paths, channel-portability commitments, and integration openness will discover the mitigations are unavailable when the migration event arrives.

Feature gaps and roadmap responsiveness. Schloter (Forbes Business Council, 17 Nov 2025) elevates responsive vendor partnership to equal weight with current product capability, on the argument that a vendor whose roadmap does not respond to the buyer's evolving needs will become a slow-motion lock-in even if the contractual terms protect against the more obvious forms. The buyer should evaluate roadmap collaboration, integration openness, and data portability with the same rigor applied to feature checklists during the initial RFP. A vendor that resists those evaluation criteria during procurement is unlikely to soften its posture after contract signature, and the buyer who fails to test for that posture before signing will be testing for it under duress later.

Acquisition risk. Gartner publishes a dedicated note on the scenario, "How to Respond When Your Conversational AI Platform Provider Is Acquired" (Rigon et al., 13 November 2025), and the existence of that note as a standalone publication is itself a signal about how frequent acquisition disruption is in the CAIP market. Acquisition disruption occurs frequently enough that Gartner treats it as a planning category rather than as an exceptional event. The response surface includes contractual protections, data portability commitments, exit clauses, and architectural decoupling, each of which must be in place before the acquisition event rather than negotiated after it.

The single procurement criterion that brings these three risks together: does the proposed partnership reduce the long-run sustainability burden, or does it merely relocate it? A platform that gets the buyer to wave two without becoming the next replatforming project passes the test. One that imports a different version of the same operating-model exposure that an in-house build would have created does not.

9. A Decision Framework for CTOs

The framework below synthesizes Schloter's seven dimensions and Gartner's nine factors (Andrews and Hare, G00825235) into a vendor-neutral comparison for conversational AI in enterprise customer service. The rows are framework-level rather than feature-level, because a feature-level table would import the bias of any one vendor's product taxonomy. The columns distinguish build, blend, and buy as three positions on the spectrum that the analyst record describes.

Criterion	Build	Blend	Buy
Strategic differentiation	Best fit when the CAI capability itself is genuinely differentiating, which is rare in defend-category CX	Platform substrate carrying common infrastructure, with proprietary logic on top	Best fit when CAI is essential infrastructure rather than competitive distinction
Sustainability capacity over five years	Requires a permanent engineering organization, with the staffing and continuity that implies	Vendor sustains the substrate while the buyer sustains the differentiating logic	Vendor absorbs the sustainment burden across the customer base
Compliance and certification maintenance	Buyer owns the full certification stack indefinitely, with audit ownership	Shared, with the vendor certifying the platform and the buyer certifying its extensions	Vendor maintains certifications across the customer base
Time to value	Longest, with the work rarely progressing past pilot without strong internal capability	Weeks to months, depending on substrate maturity and integration scope	Weeks, with vendor implementation kits and proven deployment patterns
Lock-in posture	No vendor lock-in, while foundation-model API lock-in applies if a third-party inference API is used	Substrate lock-in mitigable through AI gateways and open standards	Vendor lock-in, mitigable through data portability commitments and exit clauses
Innovation velocity	Limited by internal team capacity and by the absence of external market signal	Vendor velocity on substrate driven by cross-customer signal, buyer velocity on differentiation layer	Vendor velocity across the full stack, driven by cross-customer signal
Model quality on industry tasks	Generalist foundation-model API, with the option to invest in self-hosted fine-tuning at multi-year cost	Industry-tuned models inherited from the substrate, with the buyer's data layered onto vendor pretraining	Industry-tuned models across the full stack, refreshed on the vendor's cadence

Criterion	Build	Blend	Buy
Total cost of ownership outlook	Probability of 2x or greater budget overrun over three years for RAG specifically	Predictable substrate subscription plus scoped buyer-side investment	Predictable subscription plus integration cost
Project survival probability	Approximately 60% by 2027, given a 40% cancellation rate	Higher, with risk spread across two delivery surfaces	Highest when vendor scope is well-defined and contractual terms are robust
Replatforming horizon	Two-year shelf life for agents built before 2028	Substrate replatforming on the vendor, with buyer extensions portable if architectural discipline holds	Vendor manages model and architecture transitions

The framework answers most CTO procurement conversations directly. For defend-category customer experience, with high compliance load and a long sustainment horizon, the blend column is the correct landing position the substantial majority of the time. Build is the right call when the capability is genuinely differentiating and the engineering organization will sustain it for five years, which is a rare combination in enterprise customer service. Buy is the right call when the vendor passes the diligence bar on lock-in, data portability, roadmap responsiveness, and acquisition exposure, with the diligence conducted before contract signature rather than after.

The innovation-velocity row and the model-quality row deserve a second pass. Internal velocity is bounded by team capacity, the visible constraint, and by market exposure, the invisible constraint that §4 named. Internal model quality is bounded by the data the single enterprise can assemble, the invisible constraint that §5 named. The vendor's velocity and model quality are bounded by the same engineering constraints and amplified by cross-customer signal that the in-house team structurally cannot generate. The two columns are not comparable on staffing alone.

10. The Questions Worth Asking

Before committing to a path, the CTO should require honest answers across four categories. The questions apply equally to internal builds, blended arrangements, and external platforms, because the underlying operating-model exposures do not depend on which sourcing model carries them.

On data and compliance.

- When the system handles a customer interaction, whose infrastructure does the audio and the transcript touch at inference time, including on the ASR, NLU, dialog, NLG, TTS, and biometric paths?
- Which compliance certifications does the system carry today across PCI-DSS, SOC 2 Type II, ISO 27001, HIPAA, GDPR, and CCPA, and who audits them on what cadence?
- What is the model governance process, in terms of versioning, change control, rollback, and drift detection on every component of the conversational stack?
- What is the data residency posture in every jurisdiction the organization operates in, including the residency posture of the foundation-model provider if a wrapper architecture is in use?

On sustainability.

- Over five years, what percentage of total engineering investment goes to building versus sustaining the system across the seven components named in §2, and how many security patches, API deprecations, and regulatory updates will the organization absorb annually across those components and the six certification regimes?
- What happens when the two or three engineers who understand the system depart?
- What is the replatforming exposure if foundation-model architectures evolve materially, and what is the buyer's plan when the model the system was built on is deprecated?

On innovation velocity and model quality.

- At what cadence will the organization ship improvements, and how does that compare to a partner whose entire organization is focused on the same problem with cross-customer signal feeding the roadmap and surfacing requirements the in-house team has not yet seen?
- How many enterprise deployments will inform the system's training data and bug surfacing, and how many languages, channels, and verticals are represented in that data?
- Is the organization equipped for the Agent Development Life Cycle (ADLC) that Gartner names, or is it extending traditional SDLC into territory it does not govern?
- Will the system rely on generalist foundation-model APIs, or will it run on industry-tuned models, and if the latter, who funds the data acquisition and the fine-tuning organization required to produce them?

On lock-in and exit.

- If the architecture relies on a foundation-model provider's API, what are the options when that provider changes pricing, deprecates a model, or is acquired?
- If the architecture relies on a platform vendor, what is the data export path, the migration cost, and the realistic exit timeline?
- Are AI gateways or equivalent abstractions present in the architecture, or is the provider hard-coded?
- Do contractual terms anticipate vendor acquisition, model deprecation, and pricing change?
- The questions are identical across the three sourcing models; the honest answers are what split the path.

11. The Blend Model in Practice: The 80/20 Inversion

Gartner's blended recommendation is not a compromise position. It is the structurally correct answer for most enterprise conversational AI, and it has a specific operational shape that distinguishes it from either a pure build or a pure buy. The 80/20 inversion, named here as the central architectural figure of this paper, describes the division of labor that the blended model implies and the way in which most enterprises get the proportion wrong.

The division of labor, expressed in lifecycle-effort terms, is approximately 80/20 in the direction of platform substrate over proprietary logic. The substrate is the part of the stack that does not differentiate the buyer, and naming its components explicitly is the precondition for deciding what to buy:

- **The Conversational runtime** — STT, NLU, dialog management, NLG, and TTS running with sub-200-millisecond end-to-end latency on the voice channel, with text-channel variants that share the same intent and policy stack.
- **Channel Orchestration** — voice, chat, messaging, and email surfaces unified under a single conversation manager, with session continuity preserved across channel hops.
- **Telephony Connectivity** — SIP and RTP termination, codec handling across the carrier landscape, DTMF interpretation where the channel still demands it, and the failover discipline that voice deployments require.
- **Multilingual Coverage** — NLU and NLG support across the markets the buyer serves, with each language representing a multi-year data and tuning investment that is not a configuration toggle.
- **Voice biometrics, Spoofing, and Antifraud** — voiceprint enrollment and verification with the population scale the buyer cannot assemble on its own, paired with fraud-pattern detection trained on data the buyer does not have.
- **Conversational Analytics** — call categorization, sentiment, escalation drivers, and the reporting surface that turns the conversational system into a managed operation rather than a black box.
- **Industry-tuned Models** — smaller, faster, cheaper-to-serve models distilled and fine-tuned on the industry's cross-customer corpus, refreshed on the vendor's training cadence, which a single-enterprise build cannot match without becoming a platform vendor with one customer.
- **Compliance Certification Stack** — PCI-DSS scope for card data on voice, HIPAA scope for healthcare, GDPR scope for EU residency, SOC 2 Type II for enterprise procurement, ISO 27001 and CCPA where applicable, each maintained at the platform level rather than the buyer level.

None of those components is the buyer's competitive advantage; each is essential plumbing.

The proprietary layer, sized at roughly 20 percent of the lifecycle effort, carries the differentiation:

- **The buyer's intent taxonomy** — the catalog of customer goals specific to the buyer's products, services, and channel surfaces, which no platform can ship pre-built because no platform sees the buyer's product roadmap.

- **The buyer's escalation policy** — the rules that govern when the bot retains the conversation, when it hands off to a human, and what context the human receives, which reflect the buyer's service model rather than any platform default.
- **The brand voice** — the persona, tone, and lexical choices that distinguish the buyer's conversational system from a generic implementation, expressed in NLG prompts, dialog templates, and TTS persona configuration.
- **Integrations to the buyer's systems** — CRM, knowledge base, order management, claims handling, account servicing, with each integration representing the buyer's specific business architecture.
- **Playbooks for high-value journeys** — the multi-turn flows that the buyer's business depends on, encoded as dialog policies and tested against the buyer's actual customer interactions rather than against generic test sets.

That layer is where the buyer's competitive distinction lives, and it is the only layer where in-house engineering investment can be justified by reference to differentiation rather than to control fantasy.

The mistake to avoid is inverting the ratio. When the buyer builds the substrate and buys the logic, the organization absorbs the 90 percent of lifecycle work it cannot sustain and outsources the 10 percent that should have been its differentiator. The result is the worst position on both columns of the framework table: a high sustainment burden combined with low strategic distinctiveness. This is the inversion the figure is named for. It is the failure mode that the analyst record predicts when "build" is chosen for the wrong layer.

The first build sprint, under the correct blend, looks like this. The platform handles the runtime stack; the buyer's team owns the business logic and integration surface: intent taxonomy against the product catalog, escalation policy in the buyer's vocabulary, CRM integration built once, brand-voice NLG iterated by the CX team, and compliance scope limited to the buyer's own configuration and data handling.

The corollary is that architecture choices made at the substrate level determine portability later. AI gateways, open-standard interfaces, data export paths, modular orchestration, and decoupled retrieval layers are not engineering hygiene items; they are the difference between a blended arrangement that survives the next platform transition and one that becomes the next replatforming project (Buchanan, G00853271). The 80/20 inversion is the architecturally honest position. Anything else is a story about a capability the organization is not equipped to sustain.

12. Synthesis: Sustainability is the Test

The build-vs-buy decision is no longer principally a procurement question, because AI coding tools have democratized the creation of software while leaving untouched the governance, compliance, model operations, and five-year run-cost of operating a production conversational system across regulatory, model, and channel change. The question has become an operating-model question, in the formulation that Buchanan (G00853271) develops at length: where AI capabilities are acquired, the choice belongs in the same category as a workforce or governance decision rather than in the catalog of one-time technology purchases.

Schloter's framing and the Gartner record close on the same point. The real question is not who builds the capability but who can keep it relevant as the world changes, and three SPAs from three independent analyst teams confirm the framing with numbers.

The asymmetries of §4 and §5 sharpen the synthesis. Buildability has ceased to function as a credential; post-launch evolution has become the credential that determines the five-year outcome. The platform vendor's market exposure drives a rate of post-launch evolution and a quality of industry-tuned models that the in-house team structurally cannot match, because the engineering organization is not in the market; it is in the business, and its data is the data of one customer. The CTO who decides on buildability alone is solving the question AI coding has already answered and ignoring the two questions the analyst record centers: market exposure for roadmap velocity, and data exposure for model quality.

The practical synthesis for the CTO of an enterprise customer-service organization can be stated in four sentences. Pure build is the answer Gartner explicitly does not recommend for defend-category customer experience, which is where the substantial majority of enterprise CX automation lives (Andrews and Hare, G00825235). Pure buy is the answer only when vendor diligence on lock-in, data portability, roadmap responsiveness, and acquisition exposure has been conducted with the rigor the analyst record specifies. Blend is the structurally correct default, with platform substrate carrying the conversational runtime, channel orchestration, telephony connectivity, multilingual coverage, biometrics, analytics, industry-tuned models, and certification stack, and proprietary logic carrying the intent taxonomy, escalation policy, brand voice, integrations, and high-value journey playbooks, sized at roughly 80/20 in favor of substrate. The buyer who applies the framework rigorously will build what is genuinely differentiating and sustainable for five years, will buy what is essential infrastructure, and will architect the seam between the two with the discipline that the analyst record describes.

The test that decides the call is no longer buildability. It is the organization's exposure to the market signals that drive post-launch evolution and the data signals that drive model quality, and on those tests the partner-led blend is the structurally correct answer for the majority of enterprise conversational AI.

Sources Cited

Gartner Research

Sources are listed in the citation format Gartner specifies in its Content Compliance Policy: Gartner, [Title], [Author Name(s)], [Publication date], [Document ID where assigned].

- Gartner, “How to Decide Whether to Build, Buy or Blend Your AI Projects,” Whit Andrews, Jim Hare, 9 April 2025, G00825235.
- Gartner, “How CIOs Build, Buy and Partner to Scale AI Capabilities,” Stewart Buchanan, 28 April 2026, G00853271.
- Gartner, “Demystifying Generative AI Deployment Approaches: Critical Trade-Offs for CIOs to Navigate,” Arun Chandrasekaran, Leinar Ramos, Soyeb Barot, 2 June 2025, G00809217.
- Gartner, “Top Actions to Drive Success in Building Agentic AI Solutions,” Aaron Harrison, Haritha Khandabattu, 9 April 2026, G00846798.
- Gartner, “Select the Most Fitting Approach to Enable Conversational AI,” Gabriele Rigon, Bern Elliot, Justin Tung, Manoj Bhatia, 19 September 2025, G00839531.
- Gartner, “Magic Quadrant for Conversational AI Platforms,” Gabriele Rigon, Justin Tung, Bern Elliot, Arup Roy, Adrian Lee, Uma Challa, 13 August 2025, G00826020.
- Gartner, “How to Respond When Your Conversational AI Platform Provider Is Acquired,” Gabriele Rigon, Justin Tung, Bern Elliot, Arup Roy, Adrian Lee, Uma Challa, 13 November 2025.
- Gartner, “Toolkit: RFP Template for Conversational AI Platforms,” Gabriele Rigon, Arup Roy, Justin Tung, Uma Challa, 6 February 2026.
- Gartner, “A Buyer’s Guide for Conversational AI Platforms,” Justin Tung, Ian Elliott, Gabriele Rigon, 14 April 2026.
- Gartner, “Critical Capabilities for Conversational AI Platforms,” Justin Tung, Gabriele Rigon, Bern Elliot, Arup Roy, Adrian Lee, Uma Challa, 13 August 2025.
- Gartner, “Look Beyond Cost Savings for Lasting AI Benefits in Customer Service,” Francesco Vicchi, 2 March 2026, G00845891.
- Gartner, “Decide Between Build or Buy Solutions for RAG,” Tong Zhang, Haritha Khandabattu, Xingyu Gu, Darin Stewart, 11 September 2025, G00837989.

Other Sources

- Schloter, Philipp, “Build vs Buy in the Age of AI: A Practical Decision Framework,” Forbes Business Council (Council Post), 17 November 2025.

Gartner Trademark Notice

- Magic Quadrant is a registered trademark of Gartner, Inc. and/or its affiliates and is used herein with permission. All rights reserved.

Gartner Disclaimer

All statements in this report attributable to Gartner represent the author's interpretation of data, research opinion or viewpoints published as part of a syndicated subscription service by Gartner, Inc., and have not been reviewed by Gartner. Each Gartner publication speaks as of its original publication date (and not as of the date of this report). The opinions expressed in Gartner publications are not representations of fact and are subject to change without notice.