The Model Choice: Build, Buy or Partner across Small Models and Frontier APIs

A Transaction Framework for Control, Cost, Performance, Switching and Durable AI Economics

Authored by Chennakeshav Adya

Independent Researcher

September 2026

Abstract

Boards face a rapidly changing choice among building proprietary models or adaptations, buying a model company or application capability, and partnering with providers of open models, managed platforms and frontier application-programming interfaces. Google documents Gemma variants across mobile, laptop and server environments. Amazon Bedrock supports evaluations of foundation, customised, imported and routed models. Microsoft Foundry documents model routing as an optimisation layer that selects an eligible model for each request. These capabilities expand the feasible design space while increasing the need for disciplined economic and control choices.[1][2][3]

This paper develops a workload-to-capital framework for selecting build, buy or partner routes. It separates model access from enterprise value; maps task consequence, quality, latency, privacy, intellectual property, deployment, operating capacity and switching; and converts those variables into complete lifecycle economics. The framework treats routing, evaluation, observability, rights and exit capability as core assets rather than implementation detail.

The central decision is not whether one model is universally best. It is which combination of models, rights, infrastructure and counterparties produces accepted outcomes at an economic cost within the required control envelope. A bounded high-volume task can favour a small or specialised model. A low-volume task requiring broad capability can favour a frontier API. A strategic workflow can justify proprietary adaptation or acquisition when control and differentiation support the investment.

A hypothetical enterprise case demonstrates the method. Every task volume, quality score, price, cost, probability and valuation input is a management assumption created solely to show the calculation. It is not a forecast, quotation or representation of a named company. Six figures and seven tables connect workload classification, route scoring, lifecycle cost, switching, acquisition value and a 180-day implementation programme.

AI, privacy, intellectual property, competition, cyber-security, export, tax, accounting, valuation and investment decisions require current advice from qualified professionals. Models, provider terms, prices and regulation change. This paper provides general information for professional audiences and does not provide legal, regulatory, tax, accounting, technical or investment advice.

JEL Classification: G24, L11, L21, L24, L86, O32

Keywords: AI model strategy, small language models, frontier APIs, build buy partner, model routing, switching, AI economics, M&A, valuation

1. Frame the capital-allocation decision

Build, buy and partner are capital routes. Build commits engineering, data, evaluation and infrastructure. Buy commits acquisition consideration, integration capital and transaction risk. Partner commits the

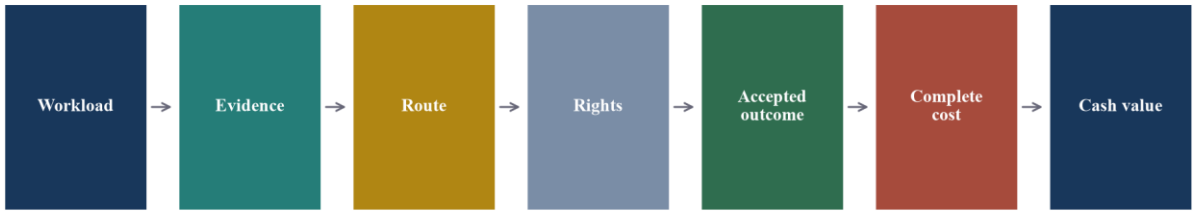
enterprise to contracts, technical interfaces and counterparty performance. The board should compare these routes against a defined workload and counterfactual.

The evidence chain begins with an eligible task, proceeds through model response and control, and ends with an accepted outcome, realised customer or operating value, complete cost and cash. Model access alone is not an asset. Value arises when the enterprise controls enough of this chain to sustain outcomes and economics.

The counterfactual must use the same workload, service level and time horizon. A build case that excludes production support cannot be compared with a partner quote that includes it. An acquisition case that credits immediate deployment cannot be compared with an internal build that assumes a long delivery period without valuing the delay. The finance team should normalise scope, timing, tax, working capital, capital expenditure and terminal obligations.

Decision rights should be assigned before a preferred supplier or target creates momentum. Product owns the business outcome. Technology owns architecture and delivery. Risk functions define the control envelope. Procurement and legal negotiate enforceable rights. Finance owns the consistent economic ledger. The investment committee decides capital and material dependency. This allocation reduces the chance that a successful demonstration becomes an unapproved strategic commitment.

Figure 1. Workload-to-capital decision chain



The chain is a governance framework and requires company-specific evidence.

2. Classify workloads before comparing models

Workload classification should precede vendor selection. Relevant dimensions include consequence of error, task variability, required context, latency, volume, data sensitivity, explainability, tool access, language, modality, offline need and human oversight. The same enterprise can have thousands of low-consequence classification events and a small number of consequential legal or credit decisions.

Classification should include the cost of abstention and escalation. A model that refuses uncertain cases can be economically attractive when a human channel exists and the residual volume is manageable. The same refusal pattern can destroy a fully automated service proposition. Evaluation should therefore measure coverage, accepted accuracy and the cost of unresolved cases together.

Task stability matters. A stable schema and narrow domain can reward investment in a smaller model, deterministic validation and specialised data. A rapidly changing knowledge task may favour retrieval and an external model whose capability evolves. A workflow with changing tools can place more value on orchestration than on the underlying model. The board should invest at the layer where differentiation can persist.

Table 1. Workload classes and plausible model routes

Workload	Primary requirement	Plausible route	Evidence gate
Repetitive classification	consistency and unit cost	small specialised model	held-out accuracy and drift
Enterprise retrieval	faithfulness and permission control	small or frontier model with retrieval	grounded answer and access test
Complex synthesis	broad capability and judgement support	frontier API with review	expert acceptance
High-volume extraction	throughput and stable schema	small model or rules plus model	field-level precision
Sensitive local task	privacy and latency	private or on-device model	data-path verification
Rare frontier task	broad knowledge and reasoning	managed frontier API	outcome quality and escalation

Routes are starting hypotheses and require evaluation against company evidence.

3. Define build, buy and partner precisely

Build can mean training a foundation model, fine-tuning an open model, creating retrieval and tools around an existing model, or developing a router and evaluation system. These paths have different capital intensity and defensibility. A board paper should name the proposed layer.

Buy can mean acquiring a model company, application, data asset, engineering team, licence or deployed customer base. The buyer needs to identify which scarce resource changes its economics. Partner can include pay-as-you-go APIs, reserved capacity, managed hosting, open-model support, joint development or distribution. Contract form determines control and switching.

Hybrid routes are common. An enterprise can build the evaluation and routing layer, buy a specialised application and partner for frontier capacity. The architecture can also change over time. Partnership can validate demand before acquisition. An acquisition can supply data and engineering while the product continues to use external models. The decision model should support staged commitments rather than force a single permanent label.

Governance should distinguish ownership from access. A purchased company may depend on third-party weights, infrastructure and licences. A self-hosted open model may still depend on a supplier for security patches and optimisation. A frontier API can offer contractual service and indemnity that exceeds an immature internal deployment. Due diligence follows the actual dependency graph.

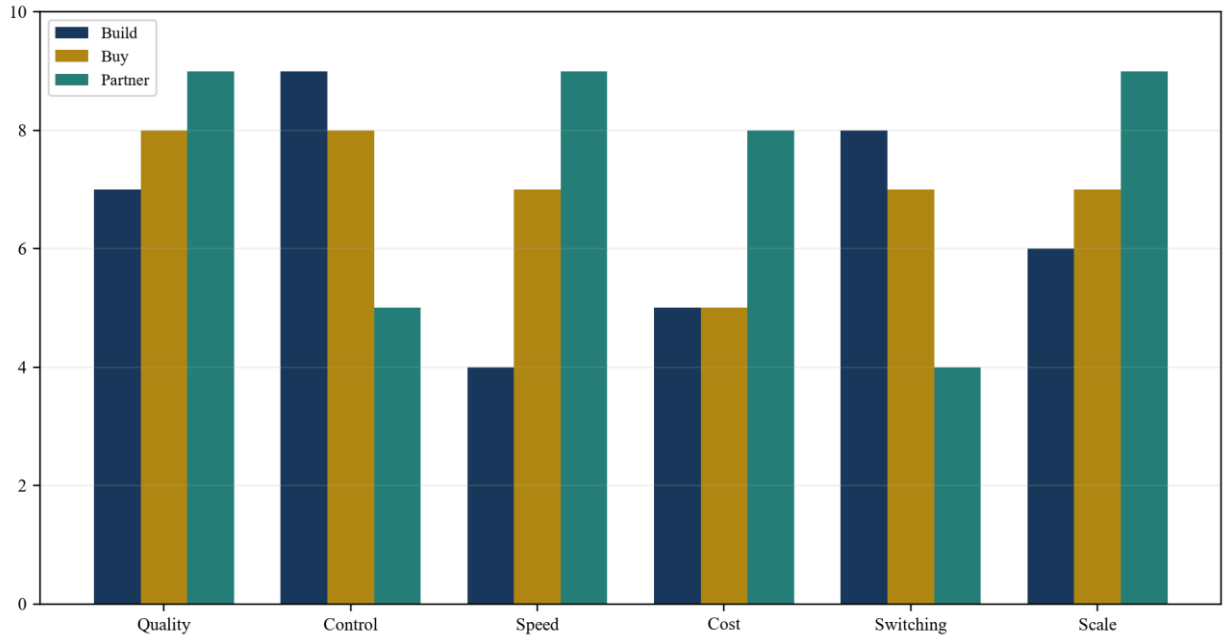
4. Establish a route scorecard

Each route should be scored on accepted quality, latency, privacy, security, intellectual-property position, deployment fit, operating resilience, complete cost, time to value and exit capability. Weights should follow the workload's consequence and strategy. A regulated workflow can place more weight on control and auditability. A consumer feature can place more weight on latency, reach and price.

Scores should be supported by evidence ranges rather than one precise number. Quality may vary by task cohort. Cost changes with volume, prompt length and fallback. Control depends on contractual rights and operating practice. The committee can assign an evidence grade beside each score and reduce the weight of weakly evidenced advantages.

A route that fails a mandatory condition should not win through a weighted average. Data residency, prohibited uses, minimum accuracy, response time, security certification or export constraints can be hard gates. The scorecard then ranks only eligible alternatives. Exceptions require named authority, duration and remediation.

Figure 2. Illustrative route scorecard



Scores are hypothetical management assumptions for method demonstration.

5. Evaluate on the enterprise task distribution

Public benchmarks help screen candidates. Investment decisions require a representative enterprise evaluation set with easy, difficult, frequent, rare and adverse cases. Amazon Bedrock documents automatic, human and model-judge evaluation and permits custom prompt datasets.[2] The enterprise should preserve ground truth, rubric, reviewer qualifications, model settings and version.

Quality should be expressed as accepted outcomes, not aesthetic preference. Measures can include correctness, completeness, faithfulness, instruction following, refusal, harmfulness and task-specific business criteria.[4] Evaluation must remain separate from training or prompt optimisation to reduce leakage.

The acceptance threshold should reflect consequence. A marketing draft can tolerate correction before publication. A payment, clinical, legal or safety workflow requires qualified review and controls appropriate to the actual decision. Aggregate scores can hide severe failure in a small but important cohort. The report should therefore show distribution, worst-case and subgroup results.

Model judges can improve scale, while they require calibration against human experts. The organisation should measure agreement, investigate systematic disagreement and retain human review for consequential cases. Evaluation data can also become a sensitive asset because it reveals customer workflows, internal rules and weaknesses. Access and retention should be controlled.

6. Reconstruct complete lifecycle cost

Token price is one component. Complete cost includes prompts and outputs, embeddings, retrieval, tools, retries, evaluation, moderation, human review, engineering, data preparation, infrastructure, observability, security, support, compliance, model updates and failed outcomes. Build also includes scarce management attention and idle capacity. Buy includes consideration, integration, retention and remediation. Partner includes minimum commitments, price changes and exit work.

Cost should be measured per accepted outcome and by cohort. A cheaper model can require more retries, longer prompts or greater human review. A capable model can reduce tool calls or raise resolution. Route

economics therefore combine price, consumption, quality and operating consequences. The ledger should reconcile provider records and infrastructure usage to customer or process events.

Fixed and variable cost affect scale differently. Internal deployment can create capacity steps and underutilisation. APIs can offer elasticity and expose the enterprise to price or currency changes. Reserved capacity can reduce unit price while increasing commitment risk. The scenario model should identify the volume at which preferred routes change and test demand volatility around that point.

Table 2. Complete lifecycle cost ledger

Cost layer	Build	Buy	Partner
Model access	compute and weights	acquired rights and licences	API or hosted charge
Adaptation	data, tuning and evaluation	remediation and integration	prompts, retrieval and tools
Production	serving, routing and monitoring	combined operating estate	usage, reservations and fallback
Control	security, privacy and governance	inherited plus buyer controls	shared-responsibility controls
Change	retraining and migration	integration and roadmap	version and provider migration
Exit	asset sale or reuse	separation or impairment	data export and replacement

The ledger requires company-specific measurement and consistent accounting policy.

7. Price control as an economic benefit

Control has value when it changes cash or risk. Proprietary deployment can support data boundaries, release timing, custom evaluation, predictable capacity and product differentiation. It also creates responsibility for reliability, security and model lifecycle. A control premium should be supported by revenue, cost avoidance, reduced loss or strategic option value.

Partnering can preserve control through data-zone terms, approved-model lists, logging, version notice, capacity commitments and termination assistance. Microsoft states that its model router respects configured geographic and compliance eligibility and discloses the selected model.[3] Company-specific verification remains necessary.

Control can also create delay. A bespoke model that reaches production a year after a partner solution loses customer learning and cash. The model should price the value of early deployment and the cost of future dependency. A staged route can launch with a partner while building portable evaluation, data and routing assets.

Board language should avoid treating control as binary. Control can reside in workload definition, data, evaluation, route, weights, infrastructure, release, customer relationship or contract. The enterprise should identify which layers are strategic and which can be supplied competitively.

8. Underwrite switching and portability

Switching is a designed capability. The enterprise should maintain model-neutral task definitions, evaluation sets, prompt and tool abstractions, exportable data, observable routes and commercial rights to migrate. Portability can be constrained by proprietary features, context formats, fine-tuning, safety systems, regional availability and customer commitments.

Migration tests should occur before dependency becomes critical. A sample of production tasks can be run through an alternative route each quarter. The test records integration effort, quality difference, latency,

cost and control gaps. A backup model that has never been tested is a procurement statement rather than operating resilience.

Exit assistance should specify data return, deletion, documentation, transition access, support and fees. Version retirement notice should provide enough time to evaluate and migrate. A right to terminate has limited value when the replacement cannot meet customer commitments.

Table 3. Switching-cost register

Dependency	Evidence	Economic effect	Mitigation
Provider-specific API	code and feature inventory	engineering and delay	abstraction and fallback
Proprietary fine-tune	rights and export test	retraining cost	portable data and evaluation
Reserved capacity	commitment schedule	stranded spend	staged commitment
Data-zone availability	regional deployment map	service restriction	eligible alternatives
Customer promise	contract review	consent or liability	route-neutral drafting
Operational knowledge	role and documentation map	continuity risk	retention and runbooks

Treatment should reflect contracts, architecture and tested migration evidence.

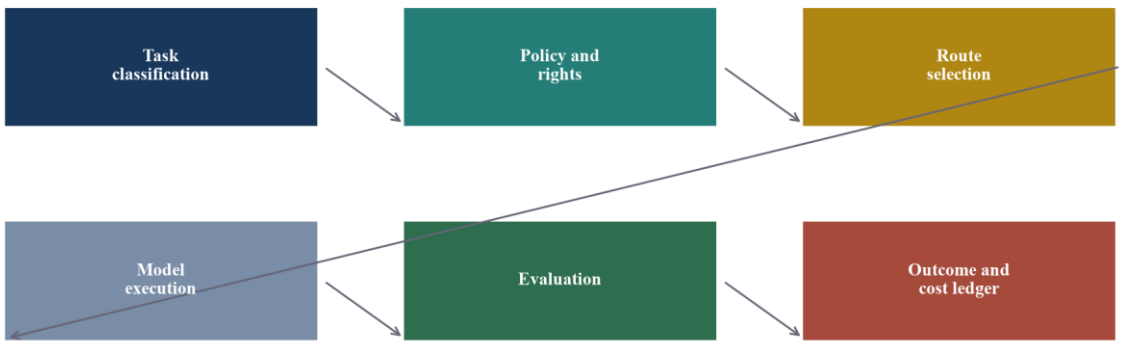
9. Treat routing as a strategic asset

A router can place simple tasks on small models and complex tasks on frontier capability. Static rules provide transparency. Learned routers can respond to prompt characteristics. Microsoft describes a lightweight routing model that selects among eligible models and reports the selected model.[3] Routing value depends on quality, latency, cost and policy evidence after routing overhead and fallback.

The router must respect eligibility before optimisation. Geography, data class, customer contract, model approval, safety and capacity can narrow the candidate set. Within that set, the route can optimise expected accepted value net of cost and latency. Policy violations should fail safely and create an auditable event.

Routing also creates model-selection data. The enterprise can learn which workloads need frontier capability and which perform well on smaller models. This evidence informs negotiation, product design and future build or acquisition. The router therefore becomes a capital-allocation measurement layer when its decisions and outcomes reconcile.

Figure 3. Model-routing control loop



The loop is a proposed operating architecture.

10. Secure model, data and output rights

Rights should cover weights, adaptations, training and evaluation data, prompts, retrieval content, outputs, telemetry, derivative work, audit and transfer. Open availability does not mean unrestricted commercial use. Google describes Gemma as a starting point that users adapt and deploy under applicable legal and responsible-use obligations.[5]

An acquisition should trace employee, contractor and third-party contributions. A partnership should state use of customer data, retention, training, confidentiality, infringement allocation and change of control. Unclear rights reduce portability and transaction value.

The diligence schedule should list every material component and its licence, owner, restriction and termination consequence. Open-source notices, model licences, data permissions and software dependencies must match the planned commercial use. A licence that works for internal evaluation can fail for redistribution or a customer-facing product.

Output rights do not resolve factual reliability or third-party claims. The operating model still needs review, attribution and use controls appropriate to the workflow. Contractual indemnity has economic value only when scope, exclusions, process and counterparty capacity are understood.

11. Build the operating capability

The operating model needs product, model, data, evaluation, security, privacy, infrastructure, finance and procurement ownership. Small models can reduce dependency while increasing responsibility for serving and updates. Frontier partnerships can reduce infrastructure burden while requiring vendor, route and usage governance.

The board should see accepted outcomes, route mix, complete cost, incidents, provider concentration, version changes and migration readiness. Technical metrics without contract and cash reconciliation are insufficient.

Release governance should bind a model version to its evaluation, prompt, tools, retrieval data, policies and infrastructure. A change in one component can alter quality and cost. Canary tests, rollback and incident authority reduce the risk of invisible production drift. Material changes should trigger customer or regulatory processes where required.

Talent strategy differs by route. Build requires durable engineering and research capacity. Buy requires retention and integration of scarce teams. Partner requires strong product, architecture, evaluation and commercial management even when model operations are outsourced. The board should avoid assuming that procurement removes the need for internal competence.

12. Govern safety, privacy and cyber-security

The NIST AI Risk Management Framework and Generative AI Profile provide structured risk-management resources.[6][7] Controls should follow the actual workload, consequence and jurisdiction. Relevant issues include data leakage, prompt injection, insecure tools, harmful output, model extraction, supply-chain compromise and unreliable automation.

Build, buy and partner allocate responsibility differently. The enterprise retains accountability for its product and decisions. Contracts and shared-responsibility statements should match the technical architecture and incident process.

Privacy review should map the complete data path, including prompts, retrieval, tools, logs, human review and support. Retention and training settings should be verified technically and contractually. Data minimisation and purpose limitation can reduce both risk and cost. Sensitive workflows may require private or local deployment, subject to actual regulatory advice.

Security testing should include the orchestration layer and connected tools. A safe model can still execute an unsafe action through excessive permissions. Identity, least privilege, transaction limits, approval and logging should sit around consequential tools. Acquired products should be tested against the buyer's threat model before integration expands access.

13. Model concentration and counterparty exposure

Provider concentration can affect price, capacity, geographic availability, model continuity and negotiating leverage. Diversification has costs because models differ in behaviour, interfaces and controls. The decision should compare the expected loss from dependency with the cost of maintaining alternatives.

Counterparty review includes financial capacity, service history, security, subcontractors, data processing, intellectual property, sanctions, export controls and termination support. A small-model partner can also create concentration when it supplies unique weights or expertise.

Concentration should be measured at several layers. Multiple applications can appear diversified while using the same model, cloud region, accelerator supplier or identity service. The enterprise should map common failure and commercial dependencies. A portfolio limit can apply to critical workloads, spend or revenue exposed to one dependency.

Financial exposure includes committed spend, prepayments, credits and transition cost. Operational exposure includes throughput, latency and service recovery. Strategic exposure includes roadmap influence and access to new capability. The board can accept concentration where benefits are strong, provided liquidity, service and migration plans match the downside.

14. Decide when acquisition creates value

Acquisition can be justified when the target owns scarce capability, data rights, distribution, deployed workflows or specialist teams that materially accelerate an approved strategy. The counterfactual should estimate internal build, partnership and delay. The buyer should separate target stand-alone value from buyer-owned distribution and infrastructure synergies.

Competition analysis can consider model access, data, ecosystems, foreclosure and innovation under current merger rules.[8][9][10] Transaction counsel should assess the actual jurisdictions and structure.

The acquisition thesis should name the integration unit. A model team may need autonomy to preserve research velocity. A product may need rapid integration with distribution and customer systems. A data asset may require strict purpose and access controls. The buyer should connect organisation design to the asset that creates value.

Technical diligence should reproduce claimed quality and cost. Commercial diligence should test customer dependence on the acquired model and the willingness to remain after control changes. Financial diligence should normalise cloud credits, capitalised development, founder compensation and shared infrastructure. Legal diligence should test model, data and code provenance and change-of-control terms.

Talent concentration can be material when a few individuals hold undocumented knowledge. Retention arrangements need knowledge transfer, succession and institutional documentation. Employment incentives should align with integration outcomes and sustainable operating evidence rather than a short demonstration milestone.

15. Hypothetical enterprise portfolio

Consider a hypothetical enterprise processing 12 million monthly tasks across extraction, retrieval, customer support and complex analysis. Every value is a management assumption for method demonstration. It is not a forecast or market quotation.

Table 4. Hypothetical workload and route portfolio

Workload	Monthly tasks	Accepted outcome	Selected route	Complete cost per accepted outcome
Extraction	6.0m	96%	built small model	AED 0.05
Retrieval	3.5m	91%	partnered small model	AED 0.18
Customer support	2.2m	88%	routed portfolio	AED 0.42
Complex analysis	0.3m	81%	frontier API plus review	AED 6.80

All values are illustrative management assumptions.

The portfolio chooses routes by workload. It does not force a single strategic label across every task. The enterprise retains evaluation, routing, data and outcome evidence as common assets.

The hypothetical build route requires AED 16 million of initial engineering and infrastructure investment and AED 6 million of annual operating cost. The buy route requires AED 38 million of consideration, AED 7 million of integration and AED 8 million of annual operating cost. The partner route has limited initial investment and usage-linked charges. These assumptions demonstrate the calculation and do not represent market prices.

Management expects routing and evaluation to reduce frontier use for extraction and retrieval while preserving frontier access for complex analysis. The case assigns no value to unsupported quality improvement. It measures accepted outcomes by workload and applies complete cost. Benefits outside the evidence period remain in a separate scenario.

The portfolio also retains a migration budget. The enterprise tests an alternative frontier route and a second small-model stack on selected tasks. The cost reduces current contribution, while providing evidence about downside continuity and negotiating leverage. The investment committee can decide whether the resilience benefit supports that recurring cost.

16. Compare the route economics

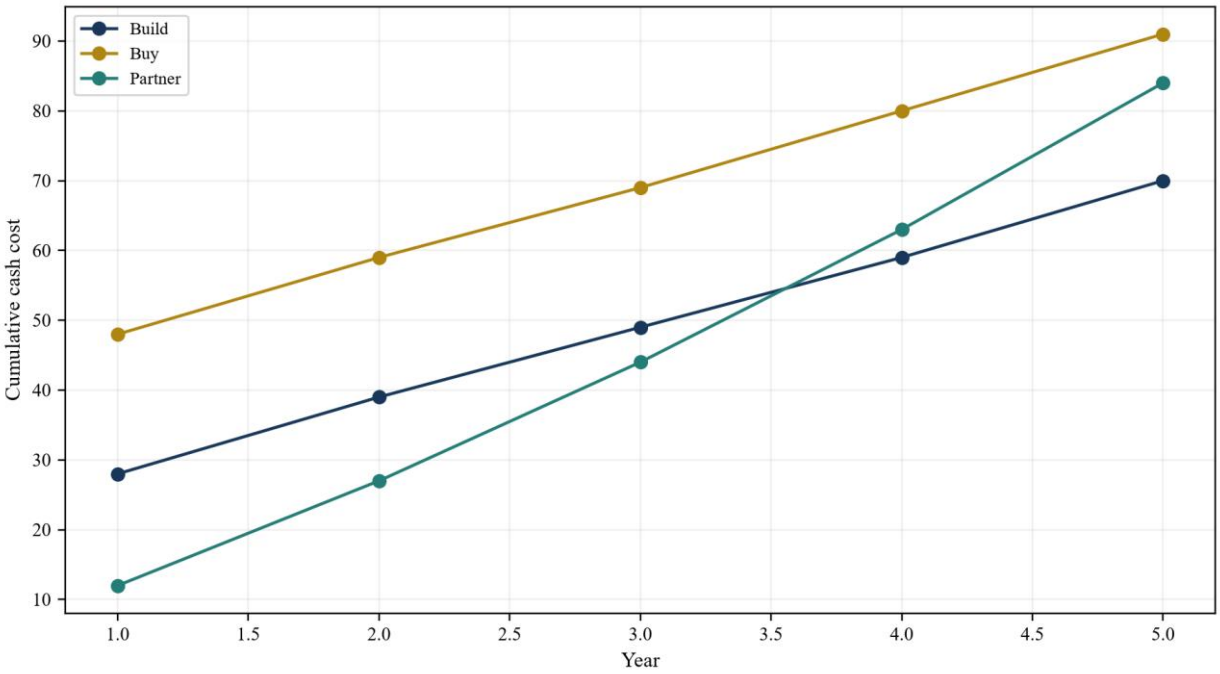
The base case should compare investment, time to value, accepted outcomes, complete unit cost, fixed and variable cost, working capital, switching and downside. Build can become economic at sufficient stable volume. Partner can remain economic when volume is uncertain or capability changes rapidly. Buy can accelerate time while adding consideration and integration risk.

Cash timing matters. A build programme can consume capital before product evidence. An acquisition can require consideration at close and retention over several years. Partner charges can scale with use and collections. The discounted-cash-flow model should reflect implementation gates, ramp, tax and working capital rather than divide a total cost by forecast volume.

Economic ownership should match accounting presentation. Gross or net revenue, capitalised development, cloud commitments, leases and acquired intangibles can affect reported metrics without changing cash in the same period. The investment committee should receive both reported-accounting and cash views under approved policy.

The break-even volume is not static. Provider prices, hardware, model efficiency and labour cost change. A build route can require periodic retraining and infrastructure refresh. A partner route can benefit from competition and new models. The board should use scenarios and decision triggers instead of one permanent crossover point.

Figure 4. Hypothetical five-year cash-cost comparison



Values are illustrative management assumptions in AED millions.

17. Stress price, quality and migration

Stress should combine provider price increases, volume growth, model retirement, quality regression, capacity restriction, security events and migration delay. Correlated events matter because a provider change can increase cost while forcing engineering and reducing quality.

The downside should flow through customer behaviour and cash. Quality regression can increase review, complaints, refunds and churn. Capacity restriction can reduce service and collections. Migration can consume engineering that would otherwise deliver product growth. The model should fund these effects and test liquidity.

Recovery time is a decision variable. A portable architecture can still require months of evaluation, remediation and customer approval. The enterprise should model degraded service and dual-running during transition. Contractual notice periods, customer obligations and regulatory approvals affect the actual timeline.

Reverse stress testing asks which combination breaks the investment case or covenant. The board then identifies early-warning measures and actions. Trigger examples include accepted-outcome decline, concentration, cost per outcome, fallback, incidents, provider notices and migration-test failure.

Table 5. Hypothetical route sensitivities

Scenario	Accepted outcomes	Annual complete cost	Migration spend	Decision implication
Base portfolio	index 100	AED 18m	AED 2m	proceed with controls
API price increase	index 100	AED 23m	AED 2m	re-route eligible tasks
Small-model regression	index 91	AED 20m	AED 3m	retrain or partner
Frontier retirement	index 94	AED 22m	AED 6m	invoke transition plan
Combined downside	index 82	AED 29m	AED 8m	capital and terms protection

Every value is an illustrative management assumption.

18. Value proprietary capability and option value

Income valuation should use cash from accepted outcomes after complete cost, investment and tax. Market references require adjustment for revenue quality, rights, concentration, growth and operating maturity. Replacement cost can inform model, data, tooling and team value while missing distribution and customer relationships.

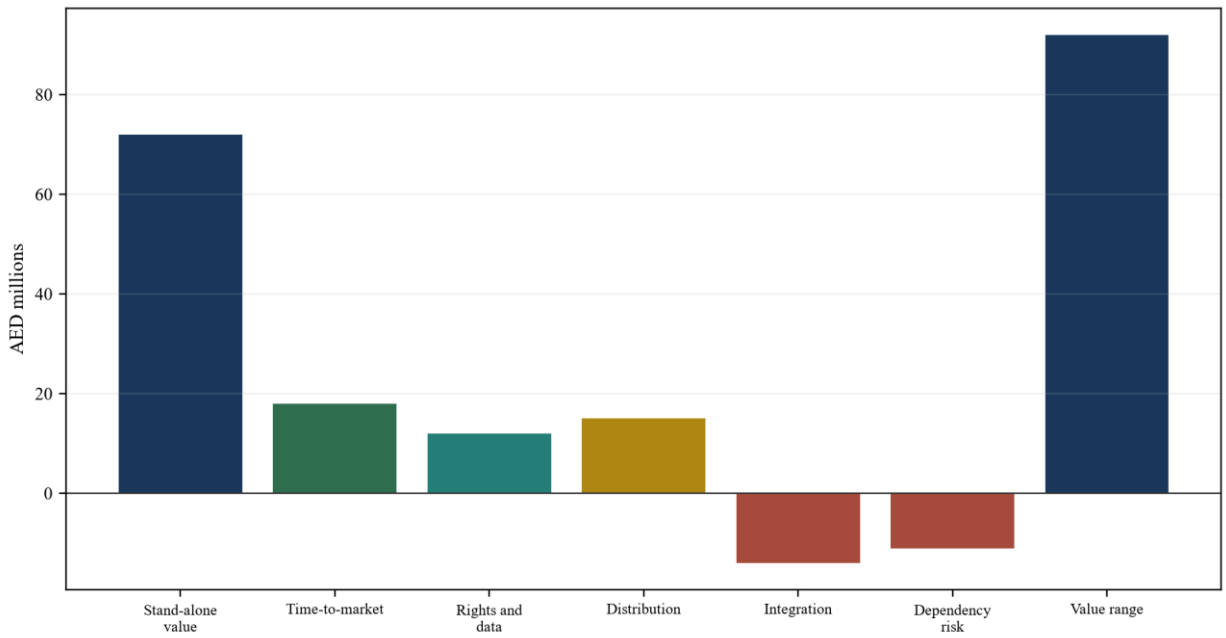
Option value can arise from rights to deploy, adapt or switch, access new workloads, or avoid dependency. The probability, investment and time required to exercise the option should be explicit. IFRS 3, IFRS 13, IAS 36 and IAS 38 govern relevant accounting and fair-value questions.[11][12][13][14]

Valuation should distinguish model capability from the operating system around it. Evaluation data, workflow integration, customer contracts, distribution and governance can remain valuable when a model changes. A model without rights, production reliability or customer access can have limited enterprise value despite strong benchmark performance.

Buyer synergies require an owner and implementation cost. Distribution synergy requires eligible customers, sales motion, price and retention. Infrastructure synergy requires compatible architecture and capacity. Data synergy requires lawful rights and technical usefulness. Paying a seller for benefits created by the buyer's existing assets overstates stand-alone value.

Contingent consideration can protect against uncertain adoption, quality or contribution. Metrics should be auditable and resistant to manipulation. Revenue can reward discounting or uneconomic usage. Accepted contribution and retention can align more closely with durable value, subject to actual legal and accounting advice.

Figure 5. Illustrative acquisition value bridge



Values are hypothetical management assumptions in AED millions.

19. Build a reproducible diligence room

Diligence should permit a reviewer to select a workload cohort, reproduce evaluation, identify model and version, trace routing and fallback, reconcile usage and direct cost, and link accepted outcomes to customer or operating value. Evidence covers code, weights, licences, data provenance, evaluations, incidents, contracts, invoices, telemetry and financial policy.

Sampling should cover important customers, languages, geographies, models, versions, workloads and failure states. Management demonstrations are starting points. Reproduction by the buyer or an independent reviewer provides stronger evidence. Exceptions should remain visible rather than being removed from averages.

The data room should maintain version history. Evaluation results can become obsolete after model, prompt, tool or retrieval changes. Financial schedules should identify the same production versions and period. This alignment supports valuation, representations and post-close measurement.

Red flags include undocumented training data, non-transferable licences, unexplained benchmark differences, dependence on promotional credits, incomplete usage reconciliation, missing incident history, customers without AI terms and provider commitments that exceed contracted demand. Each red flag needs a price, term, remediation or decline treatment.

Table 6. Minimum build-buy-partner diligence checklist

Workstream	Minimum evidence	Decision output
Workloads	taxonomy, volume and consequence	route scope
Quality	held-out set and acceptance	capability envelope
Economics	contract-to-cash and cost ledger	contribution case
Rights	model, data, code and output rights	control and portability
Operations	routing, monitoring and incidents	resilience plan
Counterparty	service, finance and concentration	dependency treatment
Organisation	roles, retention and documentation	operating model
Valuation	counterfactual, synergies and downside	price range

The checklist requires transaction-specific adaptation.

20. Allocate risk through contracts and deal terms

Partner contracts can address price, capacity, model notice, eligible regions, service, security, data, audit, transition and termination. Acquisition agreements can address rights, privacy, security, customers and financial information through warranties, covenants, indemnities, holdbacks and contingent consideration as appropriate.

Earn-outs should use auditable accepted outcomes, contribution or retention. Raw token volume can reward uneconomic use. Integration and migration funding should sit beside purchase price and provider commitments.

Pricing terms should address input, output, caching, tools, fine-tuning, hosting, support, taxes and currency. Volume tiers should be tested against downside demand. Most-favoured or benchmark provisions require enforceable definitions. Capacity and service credits should reflect the consequence of failure without creating false comfort.

Model-change provisions should define notice, documentation, evaluation access and continuation. The enterprise may require a period of parallel availability. Data and logs should remain exportable. Assistance fees and transition capacity should be agreed before termination risk rises.

Acquisition closing conditions can address critical rights, customer consent, key staff and remediation. Escrow or holdback can support specific exposures. Warranty and indemnity insurance changes recourse and does not replace evidence. Transaction advisers should tailor terms to the actual structure.

21. Execute a 180-day portfolio programme

The first thirty days establish workload, route, rights, evaluation and cost baselines. Days 31 to 90 test candidate routes, negotiate terms, remediate material controls and implement observability. Days 91 to 180 scale only workloads that pass quality, control and contribution gates.

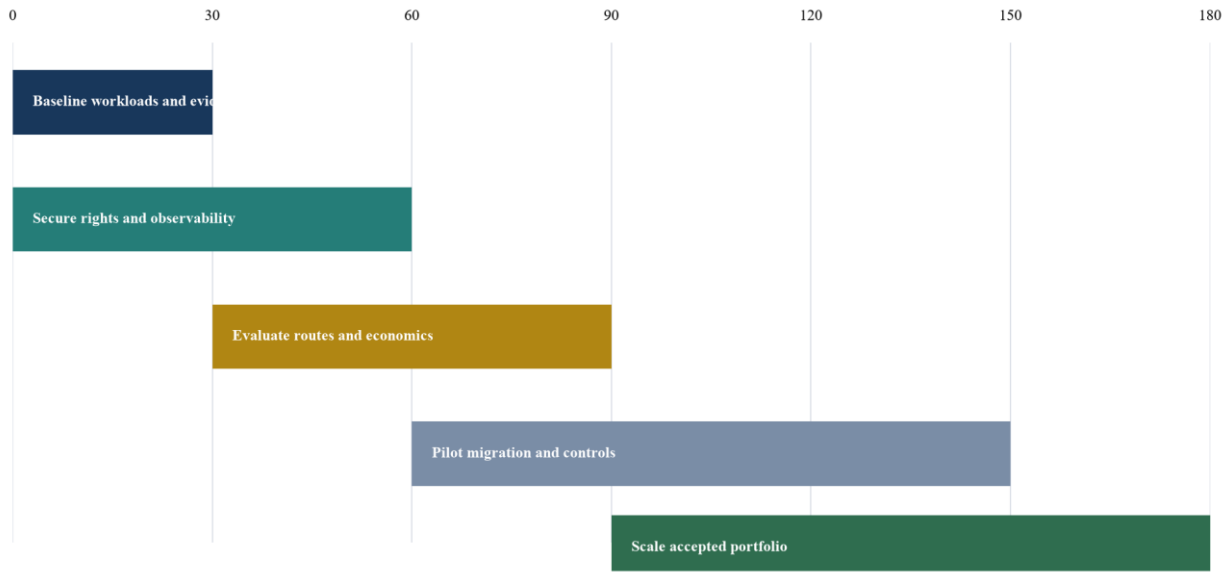
The programme needs one executive owner and a cross-functional control room. Weekly review covers delivery, quality, cost, incidents, rights and dependencies. Monthly investment review reconciles actual cash and accepted outcomes to the case. Decisions and exceptions are recorded with owners and expiry dates.

The first phase should protect continuity. The enterprise inventories production models, versions, prompts, tools, data, contracts and customer commitments. It freezes unsupported expansion and resolves critical access or security issues. Key staff and counterparties receive clear operating authority.

The second phase creates comparable evidence. Teams run the shared evaluation set, implement route telemetry, reconcile complete cost and test migration. Commercial and product teams align claims and pricing with the verified capability. Procurement negotiates options using observed demand and alternatives.

The third phase scales selectively. High-confidence workloads move to the economically preferred route. Complex or consequential cases retain appropriate review and fallback. The board receives realised contribution, investment, concentration and next capital decision.

Figure 6. 180-day model-strategy programme



The sequence is a general framework.

22. Make the board decision explicit

The memorandum should state the workload, selected route, counterfactual, evidence, complete economics, rights, dependencies, capital, downside and triggers. It should distinguish decisions that can be reversed cheaply from commitments that create structural dependency.

Approval can be staged by evidence. A limited partner pilot can validate demand. A small-model build can proceed through data, evaluation, production and scale gates. An acquisition can use exclusivity, confirmatory diligence, signing and integration gates. Each release of capital should correspond to a reduction in a material uncertainty.

The recommendation should state excluded value. Unsupported customer demand, untested migration, buyer-owned synergies and unapproved future workloads should remain outside the base case. Their potential can be shown separately with requirements to move into the case.

The board should receive a clear decline condition. Examples include rights that cannot transfer, quality that fails consequential cohorts, economics dependent on permanent credits, unacceptable data use, unmanageable concentration, or integration that the organisation cannot execute. Declining an attractive demonstration can protect capital when the evidence chain does not reach cash.

Table 7. Board decision matrix

Condition	Build	Buy	Partner
Stable high-volume bounded task	strong candidate	selective	benchmark alternative
Scarce strategic asset	possible	strong candidate	interim route
Rapidly changing frontier capability	selective	cautious	strong candidate
Strict deployment control	strong candidate	strong with rights	conditional
Limited internal capability	staged	integration dependent	strong with governance
High switching requirement	architecture-led	rights dependent	multi-provider design

Thresholds require company-specific approval.

Conclusion

The model decision is a portfolio capital-allocation problem. Workloads differ in consequence, variability, volume and control. Small models, acquired capability and frontier APIs therefore belong in a governed combination rather than a universal hierarchy.

Durable value comes from accepted outcomes, complete economics, clear rights, operating capability and tested switching. Enterprises that own evaluation, routing, data and outcome evidence can change models while preserving the product and investment thesis.

Build offers the highest potential control at the layers the enterprise genuinely owns and operates. Buy can accelerate access to scarce capability, teams, data or distribution. Partner can provide speed, frontier capability and elasticity. Each route carries obligations that need capital, governance and evidence.

The practical answer is often staged and hybrid. The enterprise builds the control plane and evidence assets, partners where capability and flexibility matter, and buys where a scarce asset materially changes time, cash or strategic position. The investment committee should revisit the route as workload, technology, price and regulation evolve.

A durable model strategy preserves the ability to learn. Every production route should produce comparable evidence about task, version, quality, latency, cost, escalation and customer response. That evidence permits the enterprise to negotiate with providers, improve smaller models, identify acquisition gaps and stop investments that do not create accepted value. Learning capacity can be more valuable than an early commitment to one model family.

Finance and technology teams need a common unit. Cost per token supports invoice control. Accepted outcome per customer or process supports capital allocation. The reconciliation between them explains why a more expensive model can be economic for one workload and wasteful for another. It also prevents reported automation volume from being treated as value before quality, price and cash are demonstrated.

Rights and switching determine how much of the operating evidence remains with the enterprise. Evaluation sets, customer relationships, workflow design, routing logic and outcome history should be

governed as strategic assets. Their ownership can preserve bargaining power even when model access is purchased. An acquisition that fails to secure these assets may transfer employees and code while leaving the economic control elsewhere.

The route also shapes financing. Predictable contracted contribution can support debt capacity, subject to lender diligence and downside coverage. Experimental workloads, concentrated providers and material migration exposure can require equity or staged capital. Acquisition debt should use cash after integration, minimum provider commitments, security remediation and transition reserves. Financing structure should reflect the same evidence used for the strategic decision.

Risk governance should remain proportionate to consequence. A low-value drafting aid can use broad experimentation within privacy and security policy. A model that triggers payments, commitments or safety actions needs stronger validation, approval, limits and monitoring. Applying one control standard to every workload either raises cost unnecessarily or leaves consequential activity under-controlled.

Model choice should remain visible after deployment. Monthly management reporting can track route share, accepted outcomes, complete cost, provider notices, migration tests and concentration. Quarterly board review can compare realised economics with the approved case and decide the next commitment. An unexpected price change, model retirement, capability shift, incident or regulatory development can reopen the build, buy or partner decision.

This approach converts a volatile technology market into a sequence of governed investment decisions. It does not require the board to predict one winning model. It requires a reproducible method for comparing routes, protecting strategic assets, funding execution and changing course when evidence changes.

The method also creates a common language for transactions and operations. Sellers can present reproducible workload evidence. Buyers can compare the target with build and partnership alternatives. Lenders can test cash resilience and dependency. Management can convert the approved thesis into measurable execution. Clear definitions support accountability across each stage.

Frequently asked questions

Q: When should an enterprise build its own model capability? A: Build can be appropriate for stable, strategically important workloads when data, expertise, volume, control and expected contribution support the full lifecycle investment.

Q: Does buy mean acquiring a foundation-model company? A: Buy can include an application, specialist model, data asset, engineering team, licence or deployed customer capability. The buyer should identify the scarce resource and its economic effect.

Q: When are frontier APIs attractive? A: They can provide rapid access to broad and changing capability, especially for lower-volume or complex tasks. Contracts and architecture should address price, version, data, region, capacity and exit.

Q: Why can small models be valuable? A: Bounded models can provide deployment control, low latency and predictable economics when task boundaries, data and evaluation are strong.

Q: How should model quality be compared? A: Use representative held-out enterprise tasks, defined acceptance rubrics, expert review where needed, fixed settings and version records. Reconcile quality to business outcomes.

Q: What is the most important switching control? A: A reproducible workload and evaluation layer, supported by model-neutral interfaces, portable data, observable routing and contractual transition rights.

Q: How does model choice affect valuation? A: It changes cash flow, capital needs, rights, concentration, operating risk and option value. Valuation should use complete lifecycle economics and separate buyer synergies.

Q: Which Matchpoint practice is relevant to this work? A: This research connects to Matchpoint Partners' strategy and execution and M&A advisory work, including model strategy, technology diligence, valuation, transaction structuring and integration.

References

[1] Google, Run Gemma Content Generation and Inferences, https://ai.google.dev/gemma/docs/run

[2] Amazon Web Services, Evaluate the Performance of Amazon Bedrock Resources, https://docs.aws.amazon.com/bedrock/latest/userguide/evaluation.html

[3] Microsoft, How Model Router Works in Microsoft Foundry, https://learn.microsoft.com/en-us/azure/foundry/openai/concepts/model-router-how-it-works

[4] Amazon Web Services, Use Metrics to Understand Model Performance, https://docs.aws.amazon.com/bedrock/latest/userguide/model-evaluation-metrics.html

[5] Google, Gemma Intended Use Statement, https://ai.google.dev/gemma/intended_use_statement

[6] National Institute of Standards and Technology, AI Risk Management Framework, https://www.nist.gov/itl/ai-risk-management-framework

[7] National Institute of Standards and Technology, Generative AI Profile, https://www.nist.gov/publications/artificial-intelligence-risk-management-framework-generative-artificial-intelligence

[8] United States Department of Justice and Federal Trade Commission, Merger Guidelines, https://www.justice.gov/atr/merger-guidelines

[9] European Commission, Merger Control, https://competition-policy.ec.europa.eu/mergers/merger-control_en

[10] United Kingdom Competition and Markets Authority, Merger Assessment Guidelines, https://www.gov.uk/government/publications/merger-assessment-guidelines

[11] IFRS Foundation, IFRS 3 Business Combinations, https://www.ifrs.org/issued-standards/list-of-standards/ifrs-3-business-combinations/

[12] IFRS Foundation, IFRS 13 Fair Value Measurement, https://www.ifrs.org/issued-standards/list-of-standards/ifrs-13-fair-value-measurement/

[13] IFRS Foundation, IAS 36 Impairment of Assets, https://www.ifrs.org/issued-standards/list-of-standards/ias-36-impairment-of-assets/

[14] IFRS Foundation, IAS 38 Intangible Assets, https://www.ifrs.org/issued-standards/list-of-standards/ias-38-intangible-assets/

[15] Google, Get Started with Gemma Models, https://ai.google.dev/gemma/docs/get_started

[16] Google, Deploy Gemma with Google Cloud, https://ai.google.dev/gemma/docs/integrations/google-cloud

[17] Amazon Web Services, Bedrock or SageMaker Decision Guide, https://docs.aws.amazon.com/pdfs/decision-guides/latest/bedrock-or-sagemaker/bedrock-or-sagemaker.pdf

[18] Microsoft, Foundry Models Sold Directly by Azure, https://learn.microsoft.com/en-us/azure/ai-foundry/foundry-models/concepts/models

[19] Microsoft, Model Catalog and Collections, https://learn.microsoft.com/en-us/azure/ai-foundry/how-to/model-catalog-overview

[20] Google Cloud, Vertex AI Model Garden, https://cloud.google.com/model-garden

[21] Amazon Web Services, Amazon Bedrock Pricing, https://aws.amazon.com/bedrock/pricing/

[22] Microsoft, Azure AI Foundry Pricing, https://azure.microsoft.com/en-us/pricing/details/ai-foundry/

[23] Google Cloud, Vertex AI Pricing, https://cloud.google.com/vertex-ai/generative-ai/pricing

[24] OpenAI, API Pricing, https://openai.com/api/pricing/

[25] Anthropic, Pricing, https://www.anthropic.com/pricing

[26] National Institute of Standards and Technology, Cybersecurity Framework 2.0, https://www.nist.gov/cyberframework

[27] National Institute of Standards and Technology, Secure Software Development Framework, https://csrc.nist.gov/Projects/ssdf

[28] National Institute of Standards and Technology, Privacy Framework, https://www.nist.gov/privacy-framework

[29] European Commission, AI Act, https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai

[30] European Union, General Data Protection Regulation, https://eur-lex.europa.eu/eli/reg/2016/679/oj

[31] OWASP Foundation, Top 10 for Large Language Model Applications, https://owasp.org/www-project-top-10-for-large-language-model-applications/

[32] MITRE, ATLAS, https://atlas.mitre.org/

[33] International Organization for Standardization, ISO/IEC 42001, https://www.iso.org/standard/81230.html

[34] International Organization for Standardization, ISO/IEC 27001, https://www.iso.org/isoiec-27001-information-security.html

[35] MLCommons, AI Safety Working Group, https://mlcommons.org/working-groups/ai-safety/

[36] MLCommons, Inference Benchmarks, https://mlcommons.org/benchmarks/inference/

[37] United States Federal Trade Commission, Premerger Notification Program, https://www.ftc.gov/enforcement/premerger-notification-program

[38] IFRS Foundation, IFRS 15 Revenue from Contracts with Customers, https://www.ifrs.org/issued-standards/list-of-standards/ifrs-15-revenue-from-contracts-with-customers/

[39] International Valuation Standards Council, International Valuation Standards, https://www.ivsc.org/standards/

[40] Financial Accounting Standards Board, Accounting Standards Codification, https://asc.fasb.org/

[41] Organisation for Economic Co-operation and Development, AI Principles, https://oecd.ai/en/ai-principles

[42] Organisation for Economic Co-operation and Development, Due Diligence Guidance for Responsible Business Conduct, https://www.oecd.org/investment/due-diligence-guidance-for-responsible-business-conduct.htm

[43] United States Securities and Exchange Commission, Cybersecurity Disclosure Guidance, https://www.sec.gov/newsroom/press-releases/2023-139

[44] Cloud Security Alliance, Security Guidance, https://cloudsecurityalliance.org/research/guidance

[45] Linux Foundation, SPDX, https://spdx.dev/

[46] Open Source Initiative, Open Source AI Definition, https://opensource.org/ai/open-source-ai-definition

[47] Google, Responsible Generative AI Toolkit, https://ai.google.dev/responsible

[48] Amazon Web Services, Responsible AI, https://aws.amazon.com/ai/responsible-ai/

[49] Microsoft, Responsible AI Standard, https://www.microsoft.com/en-us/ai/principles-and-approach

[50] International Energy Agency, Energy and AI, https://www.iea.org/reports/energy-and-ai

About the Author

Chennakeshav (CK) is a corporate finance and investment banking executive with 25+ years of global experience in deal origination, structuring and execution across M&A, growth capital and corporate strategy. He has led value-creation mandates for founders, corporates and funds — bridging the boardroom view to hands-on execution and close.

His career spans Morgan Stanley, HSBC, Lloyds Banking Group, EWEC, ADQ portfolio companies and Emirates Growth Fund, across TMT, real estate, fintech, deeptech, cleantech, infrastructure and energy. He has partnered with C-suite leaders, private equity and venture funds, sovereign wealth funds and family offices to finance complex fund raises and scale-up ventures, and has led M&A due diligence, post-merger integration and business-transformation initiatives to create value.

At Matchpoint Partners he is Managing Partner, leading the firm's corporate finance, M&A and capital-raising practice. He holds an MBA from London Business School, an engineering degree from VTU and a Master of Laws (LLM, in progress) from UCL London.

An active start-up mentor, CK mentors at Techstars, DIFC FinTech Hive, Startup Grind, Founder Institute and IN5, serves as Entrepreneur Mentor in Residence (EMiR) at London Business School, and judges the Entrepreneurship World Cup.

https://www.linkedin.com/in/ckadya/

https://www.matchpoint-partners.com/team/ck-adya.html

This paper is part of a continuing series on the structure of private and alternative markets. The views expressed are the author's own. The paper is for information only, describes market structure in general terms, and does not constitute investment, legal, tax or regulatory advice or a recommendation in respect of any security, vehicle or counterparty.