

GEM CORTEX AI

– TECHNICAL AND OPERATIONAL SPECIFICATION

Core principle: In Cortex, generative AI is a governed component of an enterprise system: it is given an identity, permissions, sources, limits, measurement and a safe option to refuse or escalate.



Executive Technical Summary

- ✓ Each request is assessed in the context of identity, purpose, data classification, permitted tools and budget.
- ✓ A deterministic path takes precedence where a database, rule or API provides a more accurate, less costly and more easily auditable result than an LLM.
- ✓ Knowledge, models, tools and the control layer are separated so that they can be changed, tested and assigned operational ownership independently.
- ✓ Actions that affect an enterprise system use explicit permissions, limits, logging and, depending on the risk, human approval.
- ✓ Quality is assessed by the accepted outcome, citations and unit cost, rather than the fluency of the text.

Minimum Product Contract

- ✓ **Input:** Identity or workload, purpose, data classification, correlation ID and cost context.
- ✓ **Decision:** Selected runtime path, model/tool, applied policy.
- ✓ **Result:** Answer, structured draft or action status, including sources.
- ✓ **Control:** Refusal, human approval, tool limits, timeout, rollback.
- ✓ **Evidence:** Audit trail, telemetry, evaluation metric and cost allocation.



1. *Reference Architecture and Allocation of Responsibilities*

The architecture is logical and layered. Its purpose is to separate responsibilities so that identity, path selection, knowledge handling, tool invocation, model runtime and operational control can be managed independently. This facilitates security review, change testing and the contractual allocation of operational responsibilities.

- **Channels and consumers:** *Workspace, portals, API, batch. The client does not need to introduce another chat interface.*
- **Access:** *SSO, roles, user and workload identity, data profile. AI inherits the permissions context.*
- **Core & Flow:** *Intent classification, runtime path selection, policies and workflows. The same input is handled by a rule, RAG, a model or a tool.*
- **Knowledge & Tools:** *Ingestion, index, retrieval, citations, connectors and tool registries. Knowledge has its own lifecycle.*
- **Model Execution:** *Provider adapters, local inference, routing and fallback. The model is a replaceable component.*
- **Control:** *Audit, telemetry, evaluations, costs and budgets. Shared evidence for the CIO, CISO and finance.*

Data and control planes: The data plane carries the specific request and result. The control plane manages configurations, catalogues, policies and budgets. The content of prompts, documents and answers is not automatically suitable for standard application logs; telemetry is minimised and classified.

2. *Runtime Paths and Decision Logic*

The runtime path specifies the mechanism by which the result is produced. The choice should be guided by the nature of the task, the availability of supporting material, the required accuracy, permissions, latency and unit economics, rather than the model's marketing appeal. The router returns both the result and a machine-readable reason for the selected path.

- **A – Direct / no LLM:** *For a known lookup, rule or precise calculation. Output from a DB/API. Control: schema, permissions, timeout. Lowest variability; testable and auditable.*
- **B – RAG + LLM:** *For an answer requiring internal knowledge. Text with citations. Control: ACL, retrieval evaluation, abstention. RAG is not a cure for hallucinations; quality is determined by ACLs and the evaluation dataset.*
- **C – General model:** *For a general transformation without internal data. Text output. Control: data profile, filter. Reduces the risk of mistaking a general answer for a corporate position.*

- **D – Tool / Workflow:** For reading from or modifying an enterprise system. Tool result. Control: scope, approval, idempotency, audit. Writes and significant actions are subject to a limit, preview and human approval.
- **E – Batch / Document:** For a large volume of files or a long-running task. Dataset, document draft. Control: queue, checkpoint, human review. Scaling document processes outside interactive chat.

3. Types of AI and Model Execution

Cortex separates the required capability from the specific model. The largest or most expensive model is not automatically the best; classification, extraction, retrieval and synthesis have different requirements.

- **LLM:** Synthesis, summarisation, dialogue. Flexible but variable output.
- **Smaller language or classification model:** Classification, field extraction. Lower latency.
- **Embedding + reranker:** Semantic retrieval. Higher relevance, but sensitive to ACLs and poisoning.
- **OCR / multimodal model:** Scans, forms, tables. Quality depends on the image.
- **Speech:** Speech transcription. Requires consent and a secure handoff.
- **Agentic system:** A multi-step plan and tool use. Bounded autonomy only: a limited objective, limits on steps, time, cost and tools, and approvals.

Model registry and routing: Each endpoint has an owner, version, region, purpose, permitted data classification and pricing profile. Routing evaluates the minimum required capability, security profile, quality on the specific dataset and cost. A fallback must not silently change the data handling arrangements. A model change is a production change and is subject to regression evaluation.

4. Data and Knowledge Layer (RAG)

Production RAG is not a one-off upload of documents to a vector database. It is a governed source lifecycle, from approval and indexing through to updates and removal.

- **Onboarding:** Owner, purpose, classification, legal basis and ACL. A source without an owner does not enter the production index.
- **Ingest:** Antimalware, parsing, OCR. Input errors are distinguished from model errors.
- **Chunking:** Splitting by structure, meaning and revisions. Chunk size affects both recall and cost.

- **Index:** Lexical and vector indexes. Hybrid retrieval combines exact terms and semantic similarity.
- **Retrieval:** ACL, tenant, role and document revision filters. Unauthorised content must not enter the model's context.
- **Context build:** Deduplication, length limits, ordering and citations.
- **Answer:** Grounded prompt, structured output, citations and abstention.
- **Freshness/delete:** Reindexing changes, deletion and propagation of ACL changes.

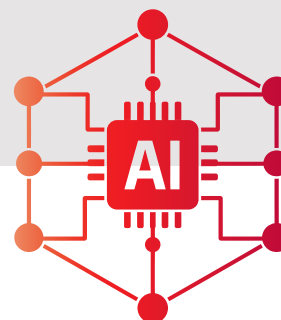
5. Integrations, APIs and the Tool Contract

Integration is the main difference between an isolated AI demo and an enterprise service. Cortex supports multiple patterns according to latency, volume, transactionality and operational ownership.



- **REST + OpenAPI:** Synchronous reads, validated writes. Required constraints: timeout, retry, version, authorisation.
- **Event / message:** Long-running workflows, batches. Required constraints: deduplication, ordering, dead-letter queue.
- **Batch / file:** Large document sets. Required constraints: manifest, checksum, checkpoint.
- **MCP (Model Context Protocol):** Standardised exposure of resources and tools. Protocol version, scopes, token audience, schema and tool risk.
- **Database read model:** Precise lookups. Read-only account, row-level security, query allowlist.
- **Webhook:** Completion notifications. Signature, replay protection, retry.

MCP nuance: MCP can unify the contract for resources and tools, but it is not, in itself, an identity provider, data catalogue or security gateway. GEM pins the supported version, validates the token audience and separates read and write tools.



6. Identities, AI Accounts and Secrets Management

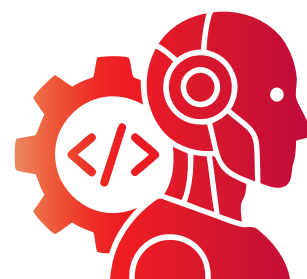
An identity-first model means that Cortex does not address security only after generating an answer. The user or workload is identified at entry, and its context is propagated to policies, retrieval and tools. This reduces the occurrence of shared accounts, unknown API keys and access rights that persist after a role change or an employee's departure.

- **User:** SSO through the enterprise IdP. The user's own corporate account, with no shared personal password.
- **Workload identity:** A separate identity for the backend, agent or connector. Separation of the person from the service.
- **Provider tenant/project:** Records of the organisation, project, region, owner and billing.
- **API key / secret:** Reference to a vault, rotation, no storage in code or prompts.
- **Target system service account:** Minimal scope, expiration, owner.

Joiner-Mover-Leaver (JML) and recertification: Onboarding creates only the required roles, a change of position recalculates permissions, and departure revokes sessions and delegations. Automation must not consolidate personal passwords. Recertification is performed according to risk.

7. Security, Privacy and Governance

The security model combines traditional application controls with risks specific to generative AI. The reference threat model is based on the OWASP Top 10 for LLMs.



- **Prompt injection:** Separation of instructions and data, sanitisation, tool policy, minimal scopes.
- **Sensitive information disclosure:** ACL before retrieval, data profile, DLP/redaction.
- **Supply chain:** Inventory, SBOM, scanning and vendor approval.
- **Improper output handling:** Schema, escaping, parameterisation and validation.
- **Excessive agency:** Bounded autonomy, budget, approval, kill switch.
- **Vector/embedding weaknesses:** Source approval, metadata, lineage, reindex.
- **Misinformation:** Citations, abstention, critical validation, human review.
- **Unbounded consumption:** Rate limit, token budget, timeout, quotas.

Data profile and provider arrangements: Each endpoint has an approved provider/data profile: purpose, region, data classification, retention, use for training and subcontractors. Zero retention or a prohibition on training may be declared only for the specific service and contractual configuration.

Governance and regulatory context: Control maintains an AI system/use-case register covering owners, purposes, models, datasets, tools, risks, changes and acceptance evidence. The evidence can support governance under NIST AI RMF, ISO/IEC 42001, GDPR, NIS2 and the AI Act. The platform itself does not automatically certify the client; the responsible organisation confirms the legal classification.

8. *Deployment Models and Their Trade-Offs*

Deployment is selected according to the data profile, network restrictions, operational capabilities, latency and economics. On-premises deployment is not automatically more secure or less expensive.

- **Customer cloud:** Infrastructure and services in the client's tenant. Control of the network, identity and costs.
- **GEM Managed cloud:** A service operated by GEM. A faster start and concentrated operational accountability.
- **Hybrid:** Data or tools internally, inference in the cloud. A combination of internal systems and flexible capacity.
- **On-prem / restricted:** Local runtime, models and data. Accommodates limited connectivity and specific data handling requirements.

Economic context: Local inference does not incur cloud token charges, but requires hardware, energy, team capacity, licences, updates and reserve capacity for peaks. TCO is therefore compared per accepted outcome and service level.

9. *AI FinOps, Budgets and Economic Telemetry*



AI FinOps links technical consumption to the owner, use case and accepted outcome. A low-cost model can create expensive rework; a higher-quality model can be more economical if its output passes review more often.

- **Allocation key:** Tenant, project, cost center, use case, environment, model, team.
- **Budget:** Monthly soft budget, alert, approval threshold and hard stop. A hard stop is safe where Cortex controls all traffic.
- **Unit economics:** Price per document, query, workflow or accepted draft.
- **Value ledger:** Baseline, volume used, acceptance rate, change in time spent, rework, TCO.

Budget management: Cortex can estimate the input size and check the budget before a call. The estimate and the final invoice may differ because of caching, batch discounts or delayed data; reconciliation is performed at financial close. A hard limit rejects further requests; a rate limit protects capacity.

Optimisation levers: Deterministic path, model routing, context, cache, batch/asynchronous mode. An optimisation is accepted only if it does not reduce the acceptance metric below the specified threshold.

10. *Non-Functional Requirements, SLO and Resilience*

A single universal availability target for all of Cortex would be misleading. A direct database lookup, interactive RAG and a batch have different characteristics. The SLO is set by runtime path and criticality.

- **Availability:** Proportion of successfully served requests.
- **Latency:** P50/P95/P99, time-to-first-token. Measure retrieval, model and tool separately.
- **Throughput:** Concurrent requests, documents per hour.
- **Quality:** Critical error rate, groundedness, draft acceptance.
- **Freshness:** Delay in source synchronisation and removal.
- **Recovery:** RTO, RPO, configuration and index backup.
- **Capacity degradation:** Fallback model, queue.
Degradation must not change the data profile.



Portability and exit: The production design includes a controlled exit: export of configurations, prompts, evaluation datasets, audit records and cost maps. It also describes what cannot be exported because of the provider's licence.

11. *Observability, Change Management and Operations*

Operating an AI service requires combining conventional observability with output quality and costs. A single correlation ID tracks a request through the gateway, retrieval, model, tool and target application.

- **Metrics:** Requests, errors, latency, tokens, cost, cache, queue, tool success.
- **Trace:** Orchestration steps, model/prompt version, router decisions.
- **Log:** Audit of access, approvals and configuration changes.
- **Eval:** Online feedback, golden set, critical errors.
- **Business event:** Accepted draft, completed case, rework.

Change management: A change to a model, prompt, data or tool requires an owner, offline evaluation, security review, staged release, approval and post-release monitoring.

Incidents and runbooks: Availability, quality, access, agent-related and financial incidents.

Reference RACI: Business outcome and process: Client (Accountable), GEM (Responsible).
Data classification: Client. Application layer: GEM. Model endpoint: Contracted operator.

Quality: GEM. Incident: Service owner.

12. *Implementation Path and Acceptance*

Implementation progresses from deciding whether the use case makes sense to demonstrating that it can be operated securely.

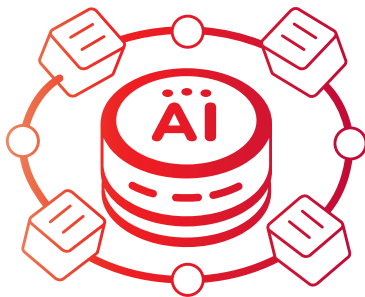
- **0. Qualification:** Business problem, sponsor, data availability, risk.
- **1. Discovery:** Baseline, data flow, integrations, evaluation dataset.
- **2. Pilot:** Limited scope, draft/read-only, measurement against the baseline.
- **3. Limited production:** Real identities, monitoring, support.
- **4. Scale:** Additional use cases, governance automation.



Acceptance pack: Functional, quality, security, operational, economic and handover acceptance. The level of autonomy is approved separately according to impact, rather than average accuracy.

13. *Limitations, Assumptions and Decision Checklist*

- **Input quality:** Poor-quality scans and conflicting documents limit accuracy.
- **Source permissions:** A connector without ACLs cannot be presented as permission-aware.
- **Provider/API:** Features, limits and prices change; adapters reduce but do not eliminate the impact.
- **Model variability:** The same input may not always produce identical text.
- **Autonomy:** Irreversible and legally significant steps remain under human control.
- **Legacy integrations:** Without documentation, integration can account for most of the effort.
- **Offline/on-prem:** Requires capacity, patching and model lifecycle management.
- **ROI:** Time savings are realised only through changes to the process and capacity.



Solution design checklist:

Are the task and baseline defined?

Are the data flow and classification known?

Are writes separated from reads?

Are there negative tests?

Are the SLO profile and exit known?

