



System Architecture and Security

Technical Architecture Document

Document ID: TR-CBX-ARCH-001

Prepared by: TechRev

For enterprise architecture and security evaluation

TECHREV

Document control

Document attribute	Value
Document identifier	TR-CBX-ARCH-001
Title	Cubix System Architecture and Security
Document owner	TechRev
Intended audience	Client enterprise architects, security teams, platform engineers and technical evaluators
Purpose	High-level technical evaluation and client-specific solution design
Scope	Logical architecture, AI integration, trust boundaries, deployment patterns and operational controls

Document conventions

Numbered sections define the architecture and its supporting controls. Identifiers TB-01 to TB-04 refer to trust boundaries; SC-01 to SC-08 refer to security controls. Bracketed reference numbers identify the external sources listed in Section 9.

Capability classifications distinguish implemented behaviour, deployment requirements and integration options. Section 1.1 defines these terms. Environment-specific implementation and acceptance evidence are addressed in the client solution design.

Document maintenance

Reassess this document when the Cubix release baseline, enabled AI features, provider interfaces or deployment design changes. Confirm external service requirements during integration. This document does not record approval of a particular client environment.

Table of contents

1. Purpose and scope	4
1.1 Capability classification	4
1.2 Document structure	4
2. Logical architecture	5
3. Trust boundaries and data flows	6
3.1 Security boundaries	6
3.2 Model-request payload	6
3.3 Data minimisation and location	6
4. AI execution and change control	7
4.1 Read and analysis sequence	7
4.2 Mutation sequence	7
4.3 Failure handling	7
5. Security control model	8
5.1 AI-specific risk treatment	8
5.2 Responsibility boundary	8
6. Deployment architecture	9
6.1 Minimum deployment elements	9
6.2 AI operating modes	9
6.3 Availability and recovery	9
7. Model-provider integration	10
7.1 Integration contract	10
7.2 Australian Government GovAI	10
8. Audit, operations and acceptance	11
8.1 Audit record classes	11
8.2 Operational requirements	11
8.3 Acceptance evidence	11
9. References and terminology	12
9.1 Product basis	12
9.2 External references	12
9.3 Terms	12

DOCUMENT SCOPE

1. Purpose and scope

This document describes the high-level technical architecture of Cubix, with emphasis on the separation between language-model inference and deterministic planning operations. It defines component responsibilities, principal data flows, security enforcement points and the deployment decisions required for a client environment.

The scope includes the Cubix application, planning engine, governed AI tools, model-provider integration and supporting operational controls. Detailed infrastructure sizing, endpoint specifications, threat modelling, recovery objectives and environment-specific security authorisation are addressed in the solution design for each deployment.

1.1 Capability classification

Implemented: a capability present in the reviewed Cubix implementation.

Deployment requirement: infrastructure or operational configuration required to establish the intended control.

Integration option: an architectural route requiring interface validation and, where necessary, adapter development. These terms distinguish product behaviour from deployment responsibilities.

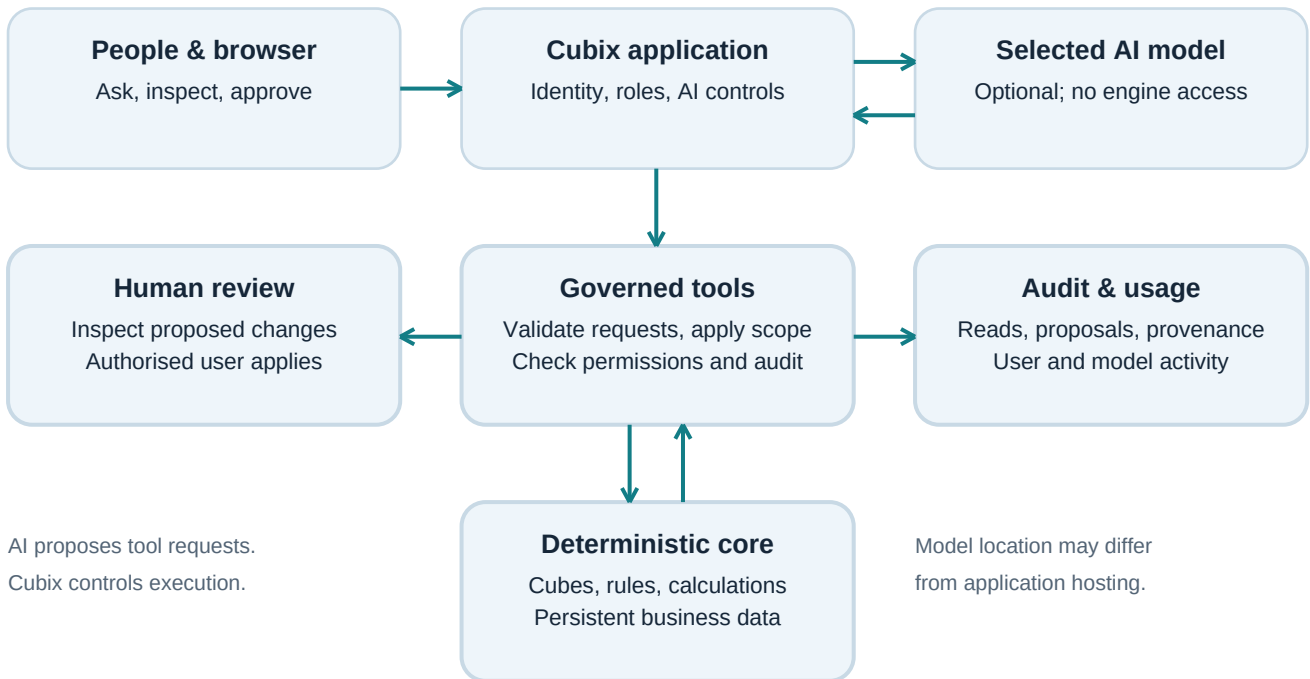
1.2 Document structure

2. Logical architecture; 3. Trust boundaries and data flows; 4. AI execution and change control; 5. Security control model; 6. Deployment architecture; 7. Model-provider integration; 8. Audit, operations and acceptance; 9. References and terminology.

SYSTEM ARCHITECTURE

2. Logical architecture

Cubix consists of a browser interface and a server application containing planning, security, integration and optional AI functions. The following diagram represents logical responsibilities; it does not imply independently deployed microservices.



Component	Responsibility
Application and identity	Authenticates requests, resolves the user principal and workspace, and exposes application and administrative functions.
AI orchestration	Builds permitted context, invokes the configured model and interprets structured tool requests. Provider credentials remain server-side.
Governed tool execution	Validates tool inputs and constrains access to the calling principal and workspace. Executes operations through scoped services rather than exposing a raw engine handle to the model.
Deterministic engine	Evaluates multidimensional models, business rules and calculations. Owns authoritative business state and executes validated changes.
Persistence and integration	Maintains model data, transaction records and snapshots. Configured connectors ingest authorised source data; audit and provenance stores support investigation.

3. Trust boundaries and data flows

3.1 Security boundaries

Boundary	Enforcement and design consideration
TB-01: Client to application	Browser traffic enters through an HTTPS endpoint. Authentication establishes the principal; the application applies workspace and role checks. The underlying Cubix HTTP listener must be reachable only through the approved access path.
TB-02: AI output to execution	Model output is treated as an untrusted request. A model-selected tool and its arguments do not confer permission. The server validates the operation and applies current authorisation before execution.
TB-03: Application to model service	The provider adapter transmits the configured prompt and result payload. This boundary may remain inside client infrastructure or cross to an approved external endpoint.
TB-04: Application to persistent storage	Business data, transaction records, conversations and audit records are retained within the deployment. Storage access, encryption, backup and retention are controlled by the selected operating environment.

3.2 Model-request payload

Depending on the AI feature, outbound requests can contain system instructions, security-filtered model metadata, user-entered text, conversation history and tool-result JSON. Tool results can include authorised business values or bounded previews. Therefore, the current architecture must not be classified as universally metadata-only processing.

The model has no direct engine or storage connection. However, tool mediation does not remove the need to approve the data included in a model request. Metadata, resource names and user prompts must also be assessed for sensitivity.

3.3 Data minimisation and location

Define permitted payloads for each enabled feature. Validate model processing location independently from application hosting, including routing, retention, logging and backup locations. A strict metadata-only mode requires additional payload restrictions and verification. For no external inference traffic, use AI-disabled operation or local inference with outbound network enforcement.

EXECUTION ARCHITECTURE

4. AI execution and change control

4.1 Read and analysis sequence

Step	Processing
1. Request admission	Authenticate the user, select the workspace and verify that AI and the requested helper are enabled for the principal.
2. Context assembly	Construct permitted schema and semantic context. Submit the user request and relevant conversation context to the selected provider.
3. Tool request validation	Interpret the returned tool request. Validate its arguments and bind execution to the authoritative user and workspace context.
4. Scoped execution	Apply current access rules and execute the deterministic operation. The model cannot expand its own resource permissions.
5. Audit and response	For governed reads, append the required audit record before releasing the result. Return permitted result data to orchestration; construct presentation blocks server-side where supported.
6. User presentation	Present the result and any AI-generated explanation. The calculated result remains authoritative; generated narrative requires user interpretation.

4.2 Mutation sequence

Structured proposal > validation > sandbox or preview > authorised human review > explicit apply > durable record. Governed AI-assisted changes use constrained proposal types. The model proposes an operation; the application validates it and the user decides whether to apply it. AI-assisted durable mutations retain provenance identifying the user, model and associated proposal or request hashes.

Human confirmation is a control point, not a universal dual-approval mechanism. Where separation of duties or independent approval is required, that workflow must be specified and validated for the intended operation.

4.3 Failure handling

Unauthorised operations are rejected. Failure to append the required governed-read audit record causes the result to be withheld. Invalid model requests and provider errors do not authorise bypassing the tool layer. AI failure does not replace deterministic calculations with generated values. Operational rollback and recovery procedures remain deployment responsibilities.

SECURITY ARCHITECTURE

5. Security control model

Control	Mechanism / requirement	Classification
SC-01 AI enablement	AI is disabled at launch by default. Without launch-time enablement, the AI layer is not loaded. A runtime master switch controls user-facing helpers after enablement.	Implemented
SC-02 AI access	Helper access is checked against the principal and administrative status. Restrict enabled helpers to approved users during rollout.	Implemented + configuration
SC-03 Data authorisation	Scoped services bind governed tool access to a workspace and principal. Authorisation is evaluated at execution; model output cannot grant additional rights.	Implemented
SC-04 Change control	Governed mutation paths require validated proposals and human application, with AI-assisted change provenance.	Implemented
SC-05 Transport protection	Terminate TLS at the reverse proxy. Restrict direct access to the HTTP listener; secure any proxy-to-application network segment according to the deployment threat model.	Deployment requirement
SC-06 Identity and privilege	Configure organisational identity/SSO, role mapping and privileged access. Enforce MFA and conditional access through the selected identity integration.	Deployment requirement
SC-07 Storage and secrets	Configure encryption for persistent storage and backups. Protect provider credentials and connector secrets; separate key material from data backups and establish rotation procedures.	Deployment requirement
SC-08 Network egress	Allow only approved provider and connector destinations. An administrative AI setting is not a substitute for network-enforced egress restrictions.	Deployment requirement

5.1 AI-specific risk treatment

Tool validation and access checks limit the effect of malicious or incorrect model instructions. They do not establish that generated output is accurate or that all prompt-injection risks are eliminated. Acceptance testing should include adversarial prompts and metadata, unauthorised resource references, excessive requests and misleading explanations.

5.2 Responsibility boundary

TechRev provides the application and agreed implementation services. The client and nominated operator establish infrastructure protection, identity policy, records retention, monitoring and workload authorisation. Responsibilities and support access must be documented before production deployment.

6. Deployment architecture

Pattern	Technical characteristics	Status
Azure	Linux container deployment with persistent storage; existing Azure deployment guidance is available. Select the client subscription, region, entry point, identity and operating model.	Documented deployment route
AWS / GCP	Deploy the application container on suitable compute with persistent storage. Implement cloud-specific networking, identity, secrets, backup and monitoring.	Deployment option; environment validation required
On-premises	Deploy on suitable client-managed server or container infrastructure. Host application data, logs and backups within the client boundary. Local model inference requires separate capacity and model validation.	Deployment option; environment validation required

6.1 Minimum deployment elements

The deployment design must identify: an HTTPS entry point; restricted application compute; persistent data storage; identity integration; protected secrets and keys; log collection; backup and restore facilities; and approved connectivity to source systems. An AI endpoint is optional and is configured separately from application hosting.

6.2 AI operating modes

AI-disabled. Do not enable the AI layer. Planning functions operate without model calls. Confirm application behaviour and outbound traffic in the acceptance environment.

Local inference. Configure a supported self-hosted provider, such as Ollama, inside the approved boundary. Include embedding services, model artefacts, hardware capacity and updates in the operating design. Enforce network restrictions to prevent unintended external routing.

Approved external inference. Use a validated provider or gateway endpoint. Approve the payload, region, authentication, retention and monitoring settings before enabling access. Application data residency alone does not determine inference residency.

6.3 Availability and recovery

Specify capacity, recovery time and recovery point objectives for the target workload. Validate snapshot-consistent backup and restoration. Multi-instance operation, replication and automatic failover must not be assumed from container packaging; the selected topology requires an explicit persistence and recovery design.

7. Model-provider integration

7.1 Integration contract

A provider integration must support the selected model, structured tool requests and responses, required context size, authentication and usage reporting. Validate error handling, cancellation and model-version changes. Endpoint compatibility alone does not demonstrate functional equivalence.

Provider route	Integration position
Ollama / Anthropic / OpenAI-compatible	Provider paths exist in Cubix. Validate the selected model and deployment configuration; an arbitrary model is not automatically interchangeable.
Microsoft Foundry on Azure	The OpenAI-compatible endpoint offers a route through the existing Cubix adapter. Validate deployment naming, authentication and tool semantics. Managed identity requires suitable token acquisition and renewal integration. [1]
Amazon Bedrock	Tool-enabled APIs offer an integration route. Select the API/model and implement or validate the adapter or approved gateway, including authentication and payload translation. This is not a prevalidated Cubix connector. [2]
Google Cloud	Gemini function calling provides an integration route through Vertex AI / Gemini Enterprise Agent Platform. Validate an adapter or compatible gateway, credentials and response mapping. [3]

7.2 Australian Government GovAI

GovAI Model Brokerage exposes approved models through an agency API subscription. Published onboarding includes API keys, outgoing IP registration and HTTPS access. Detailed interface documentation is obtained during onboarding. A Cubix integration would retain local tool execution and route model requests through the brokerage after interface validation. [4]

Current eligibility targets non-corporate Commonwealth entities and low-risk use cases. State and territory governments and corporate Commonwealth entities are not currently eligible. Engaged vendors may participate through the prescribed agency and GovTEAMS partner arrangements. Hosting and model access are separate GovAI services. [5]

Confirm selected model location, including whether global inference is permitted, and test tool semantics, usage and audit correlation. Agency workload authorisation remains necessary; platform assessment does not automatically authorise the integrated solution. [4]

8. Audit, operations and acceptance

8.1 Audit record classes

Record class	Technical purpose
Governed AI read audit	Captures timestamp, principal, workspace, conversation/turn, tool, resource, outcome, provider/model and scope/request hashes. Records omit business values. Required append failure withholds governed results.
AI mutation provenance	Associates AI-assisted durable changes with user, model, proposal or request hashes and affected destinations. Supports attribution and investigation of applied changes.
Usage accounting	Records AI usage associated with users and turns. Configure available allowances and provider quotas; application token accounting is not a guaranteed currency limit.
Conversation / diagnostic records	May contain sensitive prompts or context. Define access, retention and deletion separately from value-free audit records. Do not assume all record classes share the same retention controls.
Provider / gateway invocation logs	Required in the deployment design for complete provider-call visibility. Validate coverage and correlation with application records, including helper and internal model calls.

8.2 Operational requirements

Define log export and monitoring responsibilities, retention by record class, incident escalation and support access. Protect stored logs and verify available system audit chains. Establish patching, vulnerability remediation, secret rotation, capacity monitoring and restore testing. Prompt-content logging should be an explicit decision based on data sensitivity and investigation needs.

8.3 Acceptance evidence

Before production use, record evidence for: identity and role mapping; denied cross-workspace and restricted-data access; AI-disabled behaviour; selected provider routing; outbound payload review; proposal validation and human apply; audit-write failure handling; invocation correlation; backup restoration; and failure recovery. Record the deployed versions and configuration with the test results.

For government deployments, determine applicable AI-policy obligations, document the use-case owner and risk assessment, and complete required treatments before approval. The client-specific architecture should identify residual risks and the authority accepting them. [6]

SUPPORTING INFORMATION

9. References and terminology

9.1 Product basis

Product descriptions consolidate the Cubix logical architecture, AI architecture, security policy, deployment guidance and corresponding implementation reviewed for this document. They describe the reviewed capability set rather than a certification or an environment-specific assurance opinion. Confirm the release baseline and enabled features during solution design.

9.2 External references

Service information was checked on 10 October 2026. Reconfirm model availability, eligibility, API behaviour and commercial/data-handling terms during integration.

[1] [Microsoft Learn: Foundry endpoints](#)

[2] [AWS: Amazon Bedrock tool use](#)

[3] [Google Cloud: function calling](#)

[4] [GovAI: AI Model Access](#)

[5] [GovAI: Build with GovAI and eligibility](#)

[6] [DTA: Responsible use of AI in government](#)

9.3 Terms

Term	Meaning in this document
Deterministic core	The engine that evaluates defined model rules and calculations, independently of generated language-model reasoning.
Principal	The authenticated user or service identity against which access decisions are evaluated.
Governed tool	A server-controlled operation with validated inputs, scoped access and applicable audit requirements.
Inference	Execution of a language model to generate a response to a supplied request.
Provenance	Records associating a change with its origin, including the responsible user and AI-related context.
Trust boundary	An interface at which assumptions about identity, authority or data handling change.