CloudWinks

CLOUDWINKS · FOUNDATION APPS

CosmosCall

Technical brief

CosmosCall organizes the connections between CloudWinks and outside software. It captures how services exchange information so teams can reuse connection definitions instead of rebuilding each integration from scratch.

1. Architecture in brief

CosmosCall holds provider, endpoint/webhook, mapping-field, formatter and connection-test metadata. TheIntegrator supplies the shared execution responsibility. Request/response mapping supports synchronous calls; server execution adds a common place for controlled external HTTP execution and journaling.

2. Service boundary and intake

The connection model describes the provider, request, response and transformation needed for a call. For external error reporting, the design is outside application → CosmosCall connection → validated/mapped error → Murphy intake. The receive adapter and error contract belong to this integration boundary; an existing verifier is a reusable component rather than the whole receiving service.

3. Data handled

CosmosCall manages connection definitions, mapping fields, formatters and test metadata. KeyKook owns secret values. KeyKook connected accounts are the only connection record: CosmosCall points to them and never stores a secret itself. Catalog rows and connected accounts can each belong to a person or a company, and other people and companies cannot see them. Services that need no account share CloudWinks templates; where money or security is involved, the record belongs to the person or company and uses their own KeyKook account. Executor records describe execution outcomes. An error-reporting integration associates source application and caller context with the mapped Murphy report, keeping ownership of the error record in Murphy.

4. Integrations and ownership

Roady and MediaMogul use mapping/call services. KeyKook resolves credentials and signing secrets; TheIntegrator handles conversion and execution. Murphy receives mapped error reports in the external-service design. This separates provider connectivity, transformation, communications and error storage into clear responsibilities.

5. Security and access

CosmosCallWebhookVerifier checks HMAC signatures using KeyKook-resolved secrets, rejects missing/denied secrets and compares signatures in constant time. Execution policies control destinations and timeouts. The receiving design also includes application authorization, replay/duplicate handling, rate controls and observability; signature checking is one part of that boundary.

Competitive context

ALTERNATIVE	ITS FOCUS	CLOUDWINKS PRODUCT FOCUS
Postman	API cataloguing, design, testing and monitoring.	Connection definitions tied to CloudWinks applications and shared services.
n8n Webhook	Incoming events that start configured workflows.	A focused connection model with shared mapping and business-app context.

© 2026 CloudWinks.