

Vello System Architecture
(September 2025)

Complete Flow: Desktop → Models Gateway → Job Collector → API Service → WebSocket → Desktop

1. Ed25519 Authentication (Challenge/Verify Pattern)

- Desktop generates Ed25519 key pair + collects fingerprints

- POST /api/auth/ed25519/challenge → stores challenge in Redis (TTL 5min)

- POST /api/auth/ed25519/verify → returns Bootstrap JWT (5min, limited scope)

2. Device Registration (Bootstrap → Full JWT)

- POST /api/devices/validate → parallel fingerprint validation (collision detection)

- POST /api/devices/register → creates user/device + progressive credits (5000→1000→500→100)

- Returns Full JWT (15min) + Refresh Token (7 days)

3. Zero-Latency Transcription (Desktop → Models Gateway → OpenAI)

- POST /vello/transcribe with Full JWT + audio + session_id

- Models Gateway: ZERO JWT validation (performance) + API key injection

- Direct stream passthrough to Desktop (<0.001ms latency)

4. Async Event Processing (Models Gateway → Job Collector → API)

- Models Gateway: XADD modelsgateway:events:transcribe {metadata + session_id}

- Job Collector: Consumer group processing with DLQ retry (5s→15s→45s)

- Ed25519 authenticated POST /api/jobs/add to API Service

5. Configuration Synchronization (Job Collector ↔ API ↔ Models Gateway)

- Job Collector fetches configurations from API every 60s (Ed25519 auth)

- Writes config files and sends SIGHUP to Models Gateway for reload

- Maintains API keys, model configs, and rate limits in sync

6. WebSocket Recovery System

- Dual authentication: JWT (query param) + Ed25519 (headers)

- Orphan detection: finds Jobs without transcriptions (48h lookback)

- Automatic recovery with pending_events_batch + recovery_summary

Service Ports & Technologies:

• API Service (API_PUBLIC_BASE_URL):

FastAPI + asynccpg + PostgreSQL

• Models Gateway (MODELSGATEWAY_PUBLIC_BASE_URL):

OpenResty/Nginx + Lua + local configs

• Job Collector (JOBCOLLECTOR_PUBLIC_BASE_URL):

Python daemon + Redis streams

• Redis (REDIS_INTERNAL_BASE_URL):

Event streams + Ed25519 challenges + system prompts

• Database (DATABASE_URL):

27+ tables com Row Level Security (RLS)

Critical Security Principles:

• Desktop NEVER receives OpenAI API keys

• Zero JWT validation in Models Gateway (performance optimization)

• Ed25519 for service-to-service authentication

• Fingerprint collision detection and blocking

• Progressive credit allocation (anti-abuse)

• Session correlation via session_id cross-streams

• Time synchronization handles 2-hour drift tolerance

Desktop User

Usuário do aplicativo Desktop Vello que precisa transcrever e processar áudio usando IA

Usa aplicação para transcrever áudio

The diagram illustrates the Vello System Architecture, showing the interactions between various components within the system and with external systems.

Vello System [system]

- Job Collector Service** (Python Daemon) (JOBCOLLECTOR_PUBLIC_BASE_URL)
 - Consumer Redis streams com DLQ retry logic e Ed25519 service auth + config sync
 - 4b. Write config files + SIGHUP (openai.conf + models.conf)
 - DLQ retry logic [5s → 15s → 45s backoff]
 - Stream consumption + DLQ management
- Models Gateway** (OpenResty/Nginx + Lua) (MODELSGATEWAY_PUBLIC_BASE_URL)
 - Zero JWT validation proxy com API key injection e zero-latency streaming para OpenAI
 - 3b. Consumer group processing [transcribe/process/dlq streams]
 - 3a. XADD modelsgateway:events:transcribe [Async logging {metadata + session_id}]
 - Event streaming (no caching)
 - Store challenge + nonce (TTL 5min)
- Redis Cache & Streams** (Redis 7) (REDIS_INTERNAL_BASE_URL)
 - Event streams segregados + Ed25519 challenges + JWT cache + system prompts
- API Service** (FastAPI + Python) (API_PUBLIC_BASE_URL)
 - API principal com JWT auth, WebSocket com recovery e integração com banco via DATABASE_URL
 - 1a. POST /api/auth/ed25519/challenge [HTTPS {public_key}]
 - 1b. POST /api/auth/ed25519/verify [HTTPS {challenge_id, signature}]
 - 1c. POST /api/devices/validate [HTTPS Bootstrap JWT + {fingerprints}]
 - 1d. POST /api/devices/register [HTTPS Bootstrap JWT + {device_info}]
 - 5. WebSocket /ws/desktop [WS JWT query + Ed25519 headers]
 - CRUD operations [users, devices, credits, jobs]
 - Parallel fingerprint validation (collision detection)
 - Create user/device + progressive credits
 - Credit calculation + job creation
 - ai_models + ai_models_rules + costs
- Managed Database** (Supabase PostgreSQL) (DATABASE_URL)
 - Banco principal com RLS para 27+ tabelas: users, devices, jobs, credits
- Desktop Application** (Electron/C++)
 - Aplicação Windows que captura áudio FLAC 16kHz e gerencia autenticação Ed25519 + time sync
 - 2. POST /vello/transcribe [HTTPS Full JWT + multipart audio + session_id] Zero JWT validation <0.001ms latency
 - 1a. POST /api/auth/ed25519/challenge [HTTPS {public_key}]
 - 1b. POST /api/auth/ed25519/verify [HTTPS {challenge_id, signature}]
 - 1c. POST /api/devices/validate [HTTPS Bootstrap JWT + {fingerprints}]
 - 1d. POST /api/devices/register [HTTPS Bootstrap JWT + {device_info}]
 - 5. WebSocket /ws/desktop [WS JWT query + Ed25519 headers]
 - {challenge_id, nonce, expires_at}
 - Bootstrap JWT (5min, limited scope)
 - Full JWT (15min) + Refresh Token (7d)
 - Real-time notifications + orphan recovery

External Systems:

- OpenAI API** (Serviços de IA (Whisper, GPT) para transcrição e processamento de texto)

Legend:

- person
- container
- external system
- system boundary

Legend
person
container
external system
system boundary