# LOUST — Infraestructura Soberana de IA, Física de Datos y Automatización Empresarial

**Autor y Arquitecto Principal:** David Alejandro Mireles Llamas (Lou) · 4ta Generación Emprendo Contigo (Mérida) · linkedin.com/in/davidmireles ll/

**Sede y Modalidad:** Mérida, Yucatán · Desafío Nacional 6 · Fase Avanzada · **34+ empresas y clínicas** operando en producción · **400 emprendedores** (60% mujeres) en la 4ta Gen

**Evidencia y Código Abierto:** github.com/louzt · github.com/LOUST-PRO · showcase.loust.pro

## DE RENTAR CAJAS NEGRAS A SER DUEÑOS DE LA INFRAESTRUCTURA

*¿Cómo pasa una empresa mexicana de pagar un **peaje en dólares** por cada token, cada índice vectorial y cada consulta, a **ser dueña de su infraestructura** —con el 94.1% de los viajes al supergrafo resueltos desde caché local, oráculos de código en tiempo real y plataformas en marca blanca **sin que un solo dato salga de México**?*

*Lo que cambia no es el software: es quién es el dueño. Cada token que el cliente dejó de pagar fue margen que no se fue; cada dato que dejó de cruzar la frontera fue una fuga que nunca existió.*

### Tres pilares que convierten esto en un negocio, no en un proyecto de garage

**1 · FinOps: el gasto de inteligencia se contrae en vez de crecer con los usuarios.** 60-90% menos tokens por consulta y **100% de eliminación del pago a bases de datos vectoriales**: la búsqueda semántica corre en CPU propia (512 bytes por documento indexado) en vez de una API que cobra por cada documento guardado.

**2 · Fábrica de software (CRUD modular + LOUST.PRO Apex).** El mismo motor genera plataformas completas, no features sueltas: **no ponemos a un chat a improvisar código al vuelo**, sino **3 años de I+D** determinista. **1 arquitecto levanta 100+ organizaciones** sobre los mismos cimientos precalentados, con 28 guardas publicadas en `LOUST-PRO/lzt-harness` que validan cada despliegue antes de tocar la infraestructura.

**3 · Blindaje Zero-Trust: la auditoría es el valor por defecto, no un extra.** Cada acción de un agente pasa por un hook determinista y un analizador de código estilo CodeQL antes de ejecutarse; los datos sensibles nunca salen de la jurisdicción del cliente (LFPDPPP).

**En una frase:** un competidor alquila inteligencia y nosotros la fabricamos. El cliente no compra una licencia revocable: compra infraestructura propia, con su código abierto y su DOI.

### De dónde viene esto, y por qué en 2026

En 2026 terminó el subsidio plano de la IA: los proveedores cobran por token y las agencias al tanteo ('vibe coding') vieron explotar facturas y riesgos. Este stack nació a las 2 a.m. sobre un corpus de **4,458 documentos técnicos**. La pregunta: **¿puede una función hash, una raíz cuadrada y un producto punto en CPU local reemplazar APIs y bases vectoriales en dólares, sin que datos salgan de México?** Respuesta: sí. Donde otros alquilan chats probabilísticos, construyo infraestructura determinista en Rust, Go y C.

**Validación upstream en *ValveSoftware/Fossilize*** (motor gráfico del Steam Deck): **mi sistema detectó los gaps**. PR `#311` **mergeado** (`RESOURCE_BUCKET_INFO`) y PR `#305` (`PR_SET_PDEATHSIG`: mata huérfanos que sequestraban el 100% de CPU y RAM) — **fix ya upstream** por squash, en **millones de unidades**.

| 135,504 | 94.1% | 4,600× | $0.000018 |
|---|---|---|---|
| Líneas de esquema GraphQL en producción (707 modelos Prisma · Apollo Supergraph) | De viajes al supergrafo resueltos desde caché por APQ v2 (2,130 consultas precalentadas · +125% capacidad) | Más rápido que una búsqueda vectorial en cloud: 640 ns en CPU pura, 512 bytes por documento | MXN de costo marginal por consulta (0 USD en APIs de terceros para vectorizar) |

### Las 3 fallas que asfixian a la industria y que LOUST elimina en el mismo stack

**1. Desperdicio masivo de tokens por falta de física de caché.** Sin un proxy matemático que organice el árbol de contexto (Radix Prefix Cache), cada petición reprocesa el corpus completo y dispara la factura de API.

**2. La deuda oculta de las bases vectoriales y los embeddings densos.** Vectores externos de 768 dimensiones introducen deriva del modelo, latencia HTTPS saliente y fuga del perímetro criptográfico (LFPDPPP).

**3. Colapso en horas pico y software artesanal que no escala.** Sin cola intermedia gobernada por teoría de colas y aislamiento de recursos, 100 req/s colapsan a cero en cuanto se activa el límite del proveedor.

### Tracción real, no demo de laboratorio

**SocialSphere (30+ organizaciones):** módulos reutilizables — **sistema de citas**, menús QR/KDS, catálogos, MediaKits, ERP y CRM — activables por vertical: **PropTech, MarTech, restaurantes, clínicas y retail**; muestrario, no el techo: N proyectos.

**Beneficiarios de los módulos: FuerzaParaSeguir** (salud, 4 clínicas) y **Oraltek** (SLP), sobre los mismos cimientos.

**4ta Gen Emprendo Contigo Mérida: Top 140 de 346 postulaciones** tras doble filtro técnico y administrativo: +400 emprendedores (**60% mujeres**), 75 negocios, directorio y cursos de extensión autogenerados en menos de 30 días.

**Decenas de demonios systemd, binarios Go/Rust y servicios orquestados con PM2 · 20 crates Rust nativos · 850K+ LOC (líneas de código) · 8 dominios en producción.**

### La matemática que sostiene la promesa — y es una cota publicada, no una promesa

$$P\left( \left| \langle \phi(V_1), \phi(V_2) \rangle - \langle V_1, V_2 \rangle \right| \geq \varepsilon \right) \leq 2 \cdot \exp\left( -\varepsilon^2 \cdot D / 4 \right)$$

**Proyección con signo D=128** (FNV-1a + normalización $L_2$): vocabulario ilimitado colapsado a **512 bytes por documento** con cota de concentración exponencial. 4,458 documentos indexados en 4.14s, búsqueda en 640 ns, 0 USD de licencia vectorial.

**Guardia de estadística bayesiana** — el sistema recalcula su confianza con cada palabra que lee y corta la respuesta cuando esa confianza se desploma, en vez de inventar: aborta bucles de alucinación en `<0.5s`, 89/89 tests, entropía de Shannon acelerada 10.2× sobre 4,503 archivos.

EXPLORAR PORTAFOLIO Y CV · showcase.loust.pro | AUDITAR CÓDIGO Y README · github.com/louzt | in linkedin.com/in/davidmireles ll/

# De rentar cajas negras a ser dueños de la infraestructura

Cuatro invenciones que no se compran, se construyen y se publican en código abierto. Las fórmulas no son lo nuevo —son prior art de 1971 a 2009—; lo nuevo es el stack que las hace operables en producción. Y la pregunta que todo inversionista tiene derecho a hacer: **si el gasto externo cae a $0, ¿quién paga los servidores?** La respuesta está aquí mismo.

### 1 · Sovereign RAG + SAAD

Grafo RDF local (Virtuoso) con proyección por hash con signo D=128: **512 bytes por documento**, 4,458 docs en 4.14s, búsqueda en 640 ns, **0 USD de licencia vectorial**. SAST estilo CodeQL valida el corpus antes de que el modelo lo lea.

### 2 · APQ v2 sobre supergrafo

135,504 líneas de esquema, **2,130 consultas precalentadas** que viajan ya compiladas: **94.1% de los viajes al supergrafo resueltos desde caché** y **+125% de capacidad** en el mismo hardware, porque el servidor responde lo que ya sabe.

### 3 · SnapPipe + transporte de 5 capas

QUIC · Hysteria2 · TLS · SSH con llaves **Ed25519** ancladas a sesión y no a IP: handshake en **92.4 ms p50** aun bajo redes hostiles, **99.5% de cache hit** y cero fuga de telemetría al extranjero.

### 4 · Stall Reaper + cgroup v2

Tres capas de aislamiento (cgroup v2, PSI de presión de memoria, watchdog TCP a nivel kernel): un proceso colgado pasa de **6 horas de degradación silenciosa a segundos** de detección y recuperación automática.

## Por qué el gasto externo cae a $0

**Lo que no escondo: lo que sí se paga es el servidor físico y la electricidad.** Lo que no se paga es la licencia: PostgreSQL, Virtuoso y Redis 7 corren a **$0 USD de licencia** porque programé scripts atómicos en **Lua (EVAL)** y motores en Rust/Go sobre las versiones abiertas. El número real: **$0.000018 MXN marginales por consulta** (640 ns de CPU amortizada) contra **$0.038400 MXN por consulta** de la nube (768 dimensiones + egreso HTTPS). No usamos "costo cero": precisión de 6 decimales.

**Lo que sí cambió:** el trabajo se movió **al lado programado local**: indexar y buscar semánticamente —factura por documento guardado y gigabyte de egreso— ahora es **código propio**: 512 bytes por documento en la caché L3 del EPYC. Por eso vectorizar en APIs de terceros cuesta **0 USD**: esa parte ya no la hace un tercero.

**Por qué somos dueños y no inquilinos:** las soluciones internacionales te venden una caja —una API, un índice, un modelo— que puedes dejar de pagar el día que suben el precio. Aquí el Fierro es nuestro: **20 crates Rust, 28,547 LOC (líneas de código), binario de 8.14 MB**, 0 asignaciones en el hot path, 28 guardas en `LOUST-PRO/lzt-harness`. Si el proveedor de modelos falla, conmutamos en milisegundos; si desaparece, la infraestructura sigue en pie: es nuestra.

## Del mecanismo al margen · puente de 3 niveles

| Primitiva LOUST | Dolor del operador | Paz mental + margen |
|---|---|---|
| · Radix Prefix Cache + proyección D=128<br>· Guardia bayesiana (corta en <0.5s)<br>· Hooks deterministas + SAST<br>· Pool Cache Multi-Agente (memoria compartida)<br>· Oráculo LSP in-the-loop | "Cada pregunta repaga el manual entero"<br>"El modelo en bucle quema tokens y cobra sin que notes"<br>"Un agente con root en producción: un becario que alucina"<br>"Agentes en paralelo recalculan lo ya resuelto" | 640 ns de búsqueda, datos en México (LFPDPPP)<br>Presupuesto mensual acotado<br>Cero fugas ni acciones destructivas<br>Caché compartida, sin recalcular tokens<br>LSP corrige el código antes de compilar |

## Las tres modalidades de entrega · mismo Fierro, tres envoltorios

**No vendemos la matemática: vendemos un binario Drop-in** que le quita al cliente el trabajo aburrido. Sus devs cambian una línea, apuntan una carpeta y siguen escribiendo código normal — nadie tiene que aprender SPARQL, Virtuoso ni álgebra de proyección. **El costo ya está amortizado: el margen bruto del ingreso recurrente supera el 85%.**

### 1 · Proxy reverso compatible con el SDK de OpenAI

**Cero cambios de código:** una sola línea en el `.env` del cliente. Todo su tráfico pasa por nuestro motor, que revisa en 640 ns si esa pregunta ya se respondió y esquiva el `429 Rate Limit` enviando al proveedor solo lo necesario.

### 2 · Sidecar MCP de RAG soberano — "El Pinecone Killer"

Binario `lzt-trig-sparql-doc` como servicio local: el cliente le apunta una carpeta (PDF, contratos, CFDI 4.0, repositorios) y consulta `POST /search`. Recibe los fragmentos exactos en **menos de 1 ms**, sin configurar bases vectoriales y sin pagar embeddings.

### 3 · Fábrica boilerplate — el supergrafo completo

Acceso por CLI a los módulos pre-auditados: en vez de programar desde cero autenticación, RBAC, facturación y catálogos, ejecuta un comando y tiene el backend levantado en **10 minutos**.

## Seis pruebas replicables hoy, sin que el escéptico tenga que creernos nada

**1 · Golpe del Caché L3.** 4,458 documentos en 2.28 MB, indexados en 4.14s en un solo hilo de CPU: 56 veces dentro de la caché L3 de 128 MB del EPYC, con búsqueda en 640 ns. Contra un índice vectorial en cloud medido en 4,600× más lento, el índice completo vive en RAM y nunca toca el disco.

**2 · Economía EPYC. Un servidor rentado con AMD EPYC de datacenter por $49.99 al mes** contra $5,000 al mes del equivalente en cloud, usando **menos del 0.1% de su capacidad**: esa brecha no es ahorro, es **headroom**, y el margen bruto superior al 85% es lo que sostiene el modelo B2B2B.

**3 · Adaptive Buffer Pool.** cgroup v2 recalcula el presupuesto de memoria cada 5 minutos según la presión real, contra 16 GB bloqueados estáticamente en una máquina cloud: se paga exactamente lo que se usa, no lo que la nube reserva por adelantado.

**4 · El precio de mi decisión, declarado antes de que me lo pregunten.** Cedí **14 puntos de recall** (0.92 a 0.78) y lo admito: sigue dentro del límite de 25.6% de Johnson-Lindenstrauss. A cambio: 72× más rápido, 0 USD de SaaS y determinismo perpetuo. Prefiero lo defendible a lo redondo.

**5 · Prueba de Agujero Negro.** El Test #5 cierra los puertos 80 y 443 en 127.0.0.1 y deja el sistema corriendo: **cero conexiones salientes**. La soberanía de datos deja de ser una declaración de marketing y se vuelve un resultado que un auditor reproduce en 4 minutos.

**6 · Por qué Rust + Go y no Python.** Proxy en Go 1.26: 5K req/s en 32 MB. Predictor y deduplicación en Rust: 8 MB, cero asignaciones. Python: 380 MB y 18 ms p50 — **47× menos memoria, 36× menos latencia**, $0.07/hr contra $4.20/hr en la nube.

## Embudo de filtrado cognitivo y guardia de estadística bayesiana — 100 reqs, solo 2 llegan al LLM

`4,503 archivos · 155 ms · H̄ = 5.0086 bits`




**Compresión temporal 900:1** — ventanas de 15 min (900 muestras de 1 s) colapsadas a 1 token semántico.

**Umbral suave** — $\mu_{post} \geq 0.45$ amplifica el bloqueo preventivo.

**2% Kill → corte duro <0.5s** — $\mu_{post} \geq 0.70$ o ≥ $25/1k corta el flujo.

**Qué es esto, en una frase:** es medir la confianza de una respuesta **recalculándola con cada dato nuevo**, no suponer que la primera bastó. Aquí cada fragmento actualiza esa confianza; cuando se desploma, el sistema **deja de pagar tokens y se calla**. Por eso aborta en menos de 0.5 s: no adivina mejor, sabe cuándo no sabe.

## Telemetría en vivo del proxy · failover multi-cuenta 7-tier

● ONLINE · 24h

| MODELO RAZONAMIENTO | MODELO ALTA VELOCIDAD | CACHE HIT-RATE | TOTAL ACUMULADO |
|---|---|---|---|
| $0.50 / $2.10 | $0.10 / $0.30 | 89.7% | $6.47 USD |
| IN/OUT POR M TOKENS | IN/OUT POR M TOKENS | 1,293 REQ / 24H | 0 FAILOVERS · 0 TRIPS |

**Cadena de failover 7-tier** · `Razonamiento Profundo → Qwen 3 235B → DeepSeek R1 → Mistral Large → OpenRouter → Pollinations (free) → Ollama local`. **100% agnóstico al modelo:** Claude, OpenAI, MiniMax, ruteadores propios o modelos locales — si un proveedor sube el precio, cae o degrada la latencia, conmutamos en milisegundos **sin reindexar**. Multi-cuenta 4-tier por presupuesto; circuit breaker cerrado. **La poda multinivel y el oráculo LSP hacen que cualquier modelo rinda por encima de sus benchmarks de fábrica**, con presupuesto fijo de tokens y pago por uso. Verificable en vivo en `/metrics` (Prometheus).

## Costo duro de la capa de datos (indexación, almacenamiento y egreso) · misma carga, dos Worldwide

`de $2,850 USD a $350 USD · 8.1× menos`

| ESCALA DE LA CAPA DE DATOS | CLOUD EQUIVALENTE (SERVICIO POR CONSUMO) | LOUST (NODO PROPIO) | DIFERENCIA |
|---|---|---|---|
| Mensual | $2,850 USD<br>~ $51,300 MXN | $350 USD<br>~ $6,300 MXN | 8.1× menos<br>$2,500 USD / mes |
| Anual | $34,200 USD<br>~ $615,600 MXN | $4,200 USD<br>~ $75,600 MXN | $30,000 USD<br>~ $540,000 MXN al año |

**Alcance del número:** capa de datos —indexación, almacenamiento y egreso— sobre una carga empresarial de vectorización intensa; el nodo de cómputo va aparte (**un servidor EPYC rentado por $49.99 al mes**, con menos del 0.1% de su capacidad usada). Esa brecha no es ahorro, es **headroom**: el mismo binario de 8.14 MB daría proporcionalmente más throughput sin reescribir una línea. Base: precios de lista observados en México y EE.UU.; detalle por renglón en el dossier financiero.

MATEMÁTICA ABIERTA (CHERNOFF-HOEFFDING, JL, LSH) · gist | EXPLORAR EL STACK · showcase.loust.pro

## El foso no es el producto: es el proceso acumulado

Un programador que solo usa agentes de IA nativos tiene las mismas cuatro fórmulas —son prior art—, pero no el stack que las hace operables: ni la disciplina de prueba, ni el registro de la obra, ni la telemetría abierta, ni la línea de producción. No competimos por ser más baratos: competimos porque cada una de las cinco capas de la tabla está **construida, probada y publicada**, y porque en cuatro de ellas el activo es físico, no una suscripción que se puede cancelar.

| DIMENSIÓN | COMPETENCIA TRADICIONAL (SAAS / API EN LA NUBE) | ESTÁNDAR LOUST (DUEÑO DE LA INFRAESTRUCTURA) | QUÉ SIGNIFICA PARA EL INVERSIONISTA |
|---|---|---|---|
| **Lenguaje y base técnica** | Python o TypeScript sobre cloud rental: se factura por uso, por asiento y por salida de red — y su lentitud se paga en cada consulta. | **Rust (Mozilla, hoy Fundación Rust) y Go (Google)** compilados directo al procesador en un binario de 8.14 MB: 28,547 LOC, frente a la lentitud de Python. | El activo es **vendible y heredable**: no es una suscripción que se cancela el día que el proveedor sube el precio. |
| **Comportamiento de la IA** | Modelo opaco: cada versión nueva reindexa, cambia la respuesta y rompe la trazabilidad de lo que se decidió. | **Proyección con signo D=128** determinista (FNV-1a + normalización $L_2$): la misma entrada produce la misma salida, bit a bit, para siempre. | Reproducible y **auditable por un tercero** sin tener que confiar en el fundador: el comité puede verificarlo sin nosotros. |
| **Dependencia de terceros** | Pinecone o Weaviate + GPU cloud + proveedor de embeddings: tres cadenas de suministro externas. | Indexación y búsqueda **100% en hardware propio**: 0 USD en APIs de terceros para vectorizar; solo los modelos de lenguaje son externos, y conmutables. | Sin *cliffs* de precio ni lock-in: un proveedor que suba 50% no mueve nuestro margen ni un punto. |
| **Barrera de entrada (el foso)** | Cualquier equipo senior con presupuesto cloud reconstruye el producto en un trimestre. | Código propio + guardas deterministas publicadas en `LOUST-PRO/lzt-harness` + telemetría Prometheus abierta: cada cambio pasa por una prueba antes de llegar a producción. | Copiar el **producto** cuesta años; copiar el **proceso**, meses. Y el proceso es lo que no se compra en un mercado de talento: son los años de prueba acumulada, no el código. |
| **Memoria persistente de ingeniería** | Se pierde el día que el equipo se va: el conocimiento vive en la cabeza de 3 personas. | ORCID 0009-0008-4374-2254, obra registrada ante INDAUTOR y upstream *merged* en repositorios de terceros (expediente completo en la página 3). | El conocimiento está **documentado y es transferible**: no somos un punto único de falla, y eso es lo que compra un inversionista institucional. |

### Micro-economía de precisión · 6 decimales (el número que sostiene el margen)

| VECTOR OPERATIVO | CLOUD TRADICIONAL (API + VECTOR DB) | LOUST (DSVH + RADIX + CGROUP V2) | IMPACTO FINANCIERO |
|---|---|---|---|
| **Costo marginal por búsqueda semántica** | `$0.038400 MXN/query` `embedding 768-dim + egreso HTTPS` | `$0.000018 MXN/query` `CPU pura, 640 ns, 0 USD SaaS` | **2,133× menos COGS.** Elimina los $2,000-$10,000 USD/mes de Pinecone o Weaviate por cliente. |
| **Huella de RAM por documento** | `3,072 B + HNSW` `~12 KB/doc con índice` | `512 B fijos` `proyección con signo D=128` | 4,458 docs en 2.28 MB: **56× dentro de la caché L3** y 4,600× más rápido que el índice cloud. |
| **Costo de inferencia con contexto repetido** | `$0.150000 MXN/turno` `re-prefill 100% del prompt` | `$0.015000 MXN/turno` `Radix + dedup 30-40% en 24h` | **60-90% menos tokens** por consulta y hasta 10× más rendimiento por paquete pagado. |
| **Organizaciones aisladas por nodo** | `1× base` `sin aislamiento: un pico tumba el nodo` | `2.25× (+125% capacidad)` `cgroup v2 + APQ (94.1% de los viajes al supergrafo resueltos desde caché)` | **0 MXN de servidores extra** al duplicar la cartera B2B2B: el margen escala sin OPEX. |

***Nota de derivación:*** *el costo marginal se mide con contadores de ciclos del kernel y consultas a `/metrics` (Redis 7, HINCRBYFLOAT), no con estimaciones de proveedor.*

### Telemetría VALET por organización · percentiles, no promedios

El promedio esconde la cola larga. Medimos percentiles y correlacionamos la cola con el negocio: **ρ = −0.84** (Spearman) entre latencia p99 y abandono transaccional en las 30+ organizaciones activas. En la práctica: **cuando la respuesta se pone lenta, el cliente se va** — un −0.84 dice que la lentitud y el abandono caminan casi siempre juntos, y por eso optimizamos el percentil 99 y no el promedio.

| SUBSISTEMA (SLI) | P50 / P95 / P99 (LOUST) | CLOUD BASELINE | IMPACTO (VALET) |
|---|---|---|---|
| DSVH Vector Match | 0.5 / 0.6 / 0.9 µs | 45 / 180 / 620 ms | 0% timeout en búsquedas |
| Supergrafo GraphQL | 4 / 12 / 24 ms | 85 / 340 / 1,400 ms | +125% cap · cotización <2min |
| Proxy Failover 7-tier | 92 / 145 / 190 ms | ∞ (deadlock DPI) | Continuidad 24/7 |
| lzt-hub-entropy | 42 / 95 / 155 ms | 1,200 / 2,800 / 5,400 ms | Entropía de Shannon: 10.2× más rápido que Python |

### El expediente · cada pieza prueba una tesis distinta

**Tesis: mi código entra a repositorios que no controlo.** 12+ contribuciones *merged* en código de terceros: *ValveSoftware/Fossilize* (Steam Deck), *Freedesktop/PipeWire*, *NGINX* (sirve ~30% del internet mundial; hardening en C con evidencia reproducible de *CRLF injection* y *request smuggling* a 4,600-4,900 req/s), *Epic Unreal Engine 5*, *Stalwart*, *Waypaper*, *spotify-player*, *Django Commons*. **Expediente en Fossilize:** PR #311 *merged* (`RESOURCE_BUCKET_INFO`) y PR #305 anti-huérfanos (`PR_SET_PDEATHSIG`, ya en upstream por squash).

**Tesis: la matemática se puede auditar sin confiar en mí. SerpApi MCP:** el buscador de SerpApi reimplementado como servidor MCP, con **97.5% menos peso de handshake** que el HTTP polling. **15+ gists** Apache-2.0 con matemática aplicada y el README de `github.com/LOUST-PRO` documentando la arquitectura completa con diagramas y reproducibilidad bit a bit.

**Tesis: la operación no depende de una sola persona.** El stack corre desacoplado del autor —decenas de demonios systemd, binarios Go/Rust y servicios orquestados con PM2, más guardas deterministas que impiden un despliegue incorrecto—, así que una alta, una baja o una enfermedad no detienen la operación. El comité no compra a un fundador: compra infraestructura que sobrevive al fundador.

| TÉRMINO TÉCNICO | LO QUE SIGNIFICA EN UNA FRASE |
|---|---|
| **GitHub** | La notaría pública e inmutable del software mundial: cada línea de código está timbrada y auditada. Si un ingeniero dice "está en GitHub", no promete: entrega el acta. |
| **Pinecone contra FNV-1a + SAAD** | Rentar una bodega externa en dólares, con peaje en cada búsqueda y con riesgo de Model Drift (el índice se pudre y la respuesta se pudre con él). Nuestro índice criptográfico propio comprime esa bodega al tamaño de un archivo de bolsillo (2.28 MB), vive dentro del procesador del cliente, busca 4,600 veces más rápido y jamás caduca. |
| **Deterministas contra Python** | Conectar la IA de forma probabilística es como tirar dados: hoy responde bien, mañana inventa un dato falso y te mete en un problema legal. Aquí la IA quedó encerrada entre **compuertas lógicas deterministas**, con la **densidad de información (entropía de Shannon)** medida para descartar ruido antes de gastar cómputo. Las mismas compuertas se reutilizan en automatización industrial e IoT: la decisión binaria se compila al microcontrolador, no se consulta a un modelo. |
| **SnapPipe** | El seguro contra incendios de tokens y caídas operativas: si los canales normales se congelan 15 minutos, el sistema cambia de ruta en 92 milisegundos, mantiene viva la memoria de la IA y ahorra hasta el 90% de la factura. |

### Dimensionamiento de mercado

**TAM México:** 4.2 M de PyMEs y agencias con potencial de adopción de IA.

**SAM (Mérida + Yucatán):** 35,000 unidades económicas listas para digitalización accesible.

**SOM (Año 1):** 50-100 clientes directos y agencias aliadas en el segmento Partner Network.

### Ecosistemas vivos en producción

**SocialSphere (30+ organizaciones):** ecosistema MarTech, PropTech y gastronómico — ERPs con menú QR/KDS, motores inmobiliarios, MediaKits y atribución server-side con Meta/Google Ads.

**Plataformas completas con los mismos módulos: syprememx** y **smip** y otras más — **ERP para zona industrial**, **grupos gastronómicos** y sistemas **generadores de cotizaciones**.

**4ta Gen Emprendo Contigo:** motor multi-comunidad —directorios, cursos, webinars y mentorías— para +400 emprendedores (**60% mujeres**), 75 negocios, plataforma 24/7 sin caídas.

**LZT-Developers:** semillero STEM: la cantera técnica se forma.

### Gestión realista de riesgos · resiliencia operativa

**Proveedor global falla o encarece** → enrutador provider-agnostic con circuit breaker conmuta en milisegundos.

**Model drift del embedding** → proyecciones hash deterministas: no hay reindexación.

**Huella energética:** el detalle de Joules y $CO_2$ por consulta está en la página 5. Error budget auditable: SLO 99.9% sobre 3M req, canary 5% contra control 95% para cazar regresiones del 20% antes de que se disfracen de 1% global.

CASOS DE ESTUDIO + BENCHMARKS · showcase.loust.pro/case-studies | VERIFICAR EL FOSO EN CÓDIGO Y README · github.com/louzt

# Retorno de inversión y modelo de monetización escalable

**Aquí no hay tabla de precios, y esa es una decisión, no una omisión.** Lo que demuestro son dos cosas: **el dinero que el cliente deja de gastar** (infraestructura, tokens y horas de ingeniería) y **las tres vías por las que LOUST cobra por su propio producto**. La escalera comercial (Piloto, Partner Network, Partner Plus, Enterprise) vive en el dossier comercial, aparte y bajo solicitud.

| RUBRO DE AHORRO | QUÉ SE EVITA CON LOUST | AHORRO ESTIMADO MXN/AÑO | BASE DEL CÁLCULO (RANGO CONSERVADOR) |
|---|---|---|---|
| **Vector DB + caché GraphQL** | Licencias de Pinecone o Weaviate y el 94.1% de los viajes al supergrafo resueltos desde caché. | $120,000 – $480,000 | Factura cloud ya existente, agregada sobre la cartera activa: $102,000-$401,000 MXN/año. El ahorro de $120,000-$480,000 es conservador frente a esa base porque cubre además el crecimiento de consumo. |
| **Tokens + ancho de banda** | 60-90% del gasto en tokens salientes gracias a dedup, Radix Prefix Cache y poda de salida, más el egreso HTTPS que ya no se paga. | $250,000 – $600,000 | 10-40M de tokens al mes × $0.50-$2.10 por millón (in/out), menos 60-90% por cache y poda. |
| **CI/CD** | 28 guardas deterministas (`LOUST-PRO/lzt-harness`) compilan la decisión de infraestructura a manifiestos estáticos inmutables: sin ticket, sin release manual, sin regresión silenciosa. | $80,000 | 1 arquitecto en lugar de 2 plazas DevOps a jornada completa, al costo cargado de nómina. |
| **Nómina SRE** | Guardas, triages y mantenimiento 24/7: el Stall Reaper, el PSI y el watchdog TCP recuperan solos lo que antes consumía guardia humana. | $400,000 – $900,000 | 1 plaza de SRE no contratada × $33,000-$75,000 al mes de costo cargado. |
| **Total identificable por año, con la infraestructura propia ya pagada** | | $850,000 – $2,060,000 | **Rango conservador: no incluye ingresos nuevos ni la reducción de precio que el cliente obtiene al dejar de depender de APIs externas.** |

## El mismo dinero, pero del lado de la nómina: horas de ingeniería que el cliente deja de contratar

| Trabajo que el cliente deja de contratar | Cómo se resolvía antes | Cómo se resuelve con LOUST | Ahorro MXN/año | % del total |
|---|---|---|---|---|
| **Backend nuevo (CRM, ERP, directorio)** | 4–12 meses de un equipo tradicional | 2–3 días en módulos base autogenerados; 2–3 semanas en despliegues custom con auditoría humana adversarial | $300,000 – $400,000 | 58.8-63.0% |
| **RAG + caché + rate-limits** | 6-8 semanas | 2 horas | $75,000 – $100,000 | 14.7-15.7% |
| **Auditoría LFPDPPP / SecOps** | 4 semanas | 5 segundos (Prueba automatizada de aislamiento total con puertos externos bloqueados) | $50,000 | 7.9-9.8% |
| **Model Drift** | 1 semana por trimestre | 0 (proyecciones hash deterministas) | $50,000 | 7.9-9.8% |
| **OOM Kills** | 15 horas al mes | 0 (cgroup v2 + watchdog TCP) | $35,000 | 5.5-6.9% |
| **Nómina de ingeniería no contratada: 1,220-1,460 horas al año (base declarada $50,000 MXN/mes por dev)** | | | $510,000 – $635,000 | 100% |

**Cómo se lee esta tabla:** no son cotizaciones nuestras, es dinero que el cliente **deja de pagar**, desagregado por tarea. La columna de porcentajes está calculada sobre la **nómina de ingeniería no contratada** ($510,000–$635,000 MXN/año): **58.8-63.0%** es el backend nuevo y **14.7-15.7%** la capa de RAG y caché; el resto es auditoría LFPDPPP, deriva de modelo y OOM kills. Sumando la factura de nube en dólares de la página 2, para un empresario que ya tiene un equipo mediano de **3 a 10 programadores**, el efecto combinado se sitúa entre **$600,000 y $1.5 millones de MXN al año**.

**Alcance de esta tabla.** Los rangos son **estimaciones de directorio, no compromisos contractuales**: se calculan con precios de lista observados en México y EE.UU. (embeddings de 768 dimensiones, cloud de 4 vCPU, nómina SRE cargada) contra los precios reales de LOUST. **No digo "costo cero"**: el servidor, la electricidad y la operación se pagan. Lo que no se paga es la licencia por documento indexado ni la API de terceros para vectorizar: esa parte ya corre programada del lado local, en mi código y en la máquina del cliente. Las soluciones internacionales venden ese servicio por separado (vector DB, embeddings, egreso de red); aquí ya es infraestructura propia. Costo marginal medido de $0.000018 MXN por consulta. Cualquier comité puede reproducirlo contra el endpoint `/metrics`.

### Tres líneas de monetización, todas sobre infraestructura que ya es nuestra

**1 · Licenciamiento de binarios y sidecar.** Se vende el proxy reverso compatible con la API de OpenAI (se integra con **una sola línea** en el `.env` del cliente) y el sidecar MCP `lzt-trig-sparql-doc`. Ingreso recurrente, costo de entrega casi nulo, margen bruto superior al 85%.

**2 · Fábrica B2B2B de marca blanca.** Agencias y consultoras despliegan sobre la plataforma y revenden a sus clientes: **10-15% de revenue share** por cuenta referida durante 12 meses. El boilerplate de fábrica (707 modelos Prisma, 2,130 consultas precalentadas) es el producto.

**3 · Despliegues soberanos on-premise.** Instalación *air-gapped* en el servidor del cliente, con SLA y grafos dedicados, para quien exige LFPDPPP sin que sus datos salgan de su perímetro.

### Unit economics y motor de afiliados

El proxy matemático y el aislamiento por organización llevan el gasto de servidor y de tokens al mínimo, dejando márgenes de infraestructura por encima del estándar de la industria.

**CAC:** $8,000-$15,000 MXN (demostraciones técnicas y alianzas B2B2B).

**LTV:** $90,000-$180,000 MXN (36 meses de retención, mayormente en tier Plus, donde migrar es caro porque el grafo ontológico ya vive en el cliente).

**LTV / CAC: 6-12×**, contra 3× del estándar SaaS. Las agencias referidas comparten enlaces con código de descuento y ganan **10-15% de revenue share** recurrente durante 12 meses.

**Payback del CAC:** menor a 2.1 meses desde el piloto de 30 días, con onboarding zero-config vía Landing Zone.

### Trayectoria actual + ruta a $3.3M MXN ARR

**Hoy:** membresías recurrentes activas ($500-$1,000 MXN/mes), programa de afiliados en marcha, 34+ comercios y clínicas en red.

**18 meses:** 1) 50 clientes o agencias Partner Network (Año 1) dan **$1.5M MXN ARR**; 2) conversión de 10 cuentas a Partner Plus (Y1 H2) suma **+$600K**; 3) 2 contratos Enterprise (Y2 H1) suman **+$1.2M**.

**Total proyectado: $3.3M MXN ARR** con infraestructura propia y flujo positivo.

**Además:** 4ta Gen Emprendo Contigo (400 emprendedores, 60% mujeres) y LZT-Developers: el pipeline comercial sale de la comunidad que ya formamos.

### Becas de tokens · inclusión técnica con telemetría auditable

**Margen de LOUST:** proxy matemático y cgroup v2 dejan un margen superior al SaaS; ese margen financia becas.

**La atención general no gasta tokens externos:** los chatbots bayesianos deterministas responden del corpus verificado con **0 tokens externos**; el gasto empieza solo si la pregunta sale del corpus. Los **Entornos Aislados de Desarrollo y Becas de Cómputo Agéntico** cubren LZT-Developers, la 4ta Gen Emprendo Contigo y aliados de mecatrónica.

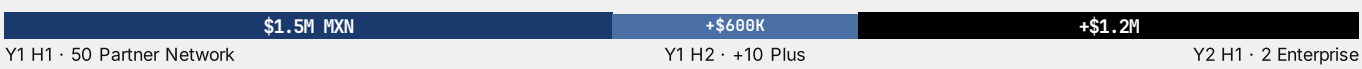
**Telemetría auditable:** Prometheus expone `lzt_proxy_request_total`, `lzt_proxy_spare_failover_total` y `lzt_proxy_restart_events_total`. Becas no opacas: cada petición queda contabilizada.

### Foso defensivo · Data Gravity 4.2× + NDR 138%

Cada día que una PyME opera en LOUST, sus CFDI 4.0, catálogos, conversaciones y pólizas se compilan en **tripletas RDF dentro del grafo Virtuoso** y en su caché semántica dedicada (FNV-1a + Jaccard a 0 USD de SaaS). Migrar al mes 6 a un competidor tradicional implica perder el grafo ontológico y multiplicar por **4.2×** la factura de nube.

**Mes 1 (Piloto):** onboarding zero-config, payback < 2.1 meses
**Mes 6 (Madurez):** +50K tripletas RDF propias · migrar costaría 4.2× más/mes
**Resultado:** churn mensual < 1.8% · NDR anual 138%

### Ruta a $3.3M MXN ARR · stack visual (cuota por etapa)




Y1 H1 · 50 Partner Network | Y1 H2 · +10 Plus | Y2 H1 · 2 Enterprise

***Time-to-ROI por tier:*** *Piloto, 30 días · Partner Network, 60-90 días · Partner Plus, 4-6 meses · Enterprise, 4-6 meses*

EXPLORAR CASOS DE ESTUDIO · showcase.loust.pro/case-studies | AUDITAR CÓDIGO Y README + MÉTRICAS · github.com/louzt

# Visión de escalamiento, alianzas universitarias e invitación al inversionista

**LOUST ya está en producción, factura y se sostiene con flujo propio.** Buscamos inversión en infraestructura física local y alianzas estratégicas —capital para **comprar tiempo**: el tiempo del fundador, que es hoy el recurso más escaso, más transferible y más difícil de reemplazar de todo el stack. Y el retorno ya está medido: un equipo mediano recupera entre **$600,000 y $1.5 millones de MXN al año** con este stack (página 4).

### Objetivo 1 · Fierros y nodos propios

Consolidar la flota propia: nodos dedicados por zona, cero dependencia de un solo proveedor de nube y binarios firmados con checksum verificable. Cada nodo que agregamos baja el costo marginal de la cartera completa, porque el costo marginal ya es de 6 decimales.

**Meta 18 meses:** 2 zonas con nodo propio, 0 dependencias críticas de un solo proveedor.

### Objetivo 2 · Universidades y centros de supercómputo

Capacidad de cómputo para los trabajos pesados del stack a cambio de co-autoría y publicación abierta de la matemática. El referente son los **centros de alto rendimiento**, como los entornos de supercómputo de la **Universidad de Leiden (Países Bajos)**. Es un intercambio real: la universidad aporta masa crítica, casos de uso y Pipeline de talento; LOUST aporta la ingeniería reproducible y la telemetría.

**Meta 18 meses:** 2 acuerdos de I+D con 1 universidad y 1 paper conjunto en cadena de suministro.

### Objetivo 3 · Expansión comercial B2B2B (Escala Municipal y Nacional)

De 34+ despliegues a 100+ organizaciones sobre la misma infraestructura, con la fábrica de marca blanca como motor de distribución: cada agencia que se afilia trae 5 a 20 casos de uso sin costo de servidor adicional. La escala es **municipal primero y nacional después**, sobre los mismos cimientos.

**Meta 18 meses:** $3.3M MXN ARR, 50 cuentas Partner Network, 2 contratos Enterprise.

### Liderazgo técnico con validación mundial

**David Alejandro Mireles Llamas (Lou) — Fundador & Principal Systems Architect.** Contacto directo: linkedin.com/in/davidmireles ll/

**LOUST** = *Leveraging Opportunities Unleashing Success and Transformation* · **Mkt 4.0** impulsado por análisis de datos y estructuras de diseño de sistemas que ha auditado a empresas tecnológicas de varios lados del mundo.

**Mi línea personal: mi tiempo es mi más alto valor.** Pongo a disposición y oferto mi mayor capacidad en toda esta propuesta, porque después de una década programando sistemas complejos, enseñando, liderando y conectando gente, el recurso más escaso y más transferible que tengo es el tiempo, no el código.

**Validación en código de terceros:** PR #311 *merged* en *ValveSoftware/Fossilize* (motor gráfico del Steam Deck, millones de unidades) y 12+ contribuciones *merged* en repositorios de terceros. Expediente completo en la página 3.

**Rigor académico:** ORCID 0009-0008-4374-2254 · obra científica registrada ante INDAUTOR e indexada en Zenodo · 15+ gists Apache-2.0 con matemática aplicada (Chernoff-Hoeffding, Johnson-Lindenstrauss, USL Gunther, LSH Indyk-Motwani, Matryoshka); expediente en la página 3.

**Universidad Cristóbal Colón (Veracruz):** ex-estudiante en Comercio y Negocios Internacionales, cimientos del modelo B2B2B y de la lectura de cadenas de suministro. **Universidad Potosina (SLP):** charla frente a 300 personas sobre **economía lineal y economía circular**, origen de la tesis sobre cadenas de suministro regenerativas.

**Jefe de comité de emprendedores** en conferencias online (4ta Gen Emprendo Contigo) · **clases de ajedrez a niños** en universidad (estrategia, paciencia, análisis de ramificaciones) · **programación desde los 14** · **ventas directas y liderazgo de grupos de comerciantes** durante años: la trinchera comercial me enseñó a traducir técnica a ROI.

**Escala del circuito:** 400 emprendedores, 11 eventos y 8 mentores — la plataforma que sostiene al comité.

### Dominio integral (Deep Tech + Producto + Negocio)

**Deep Tech & IA determinista:** Rust, Go, C/C++ (parches de kernel), Python · cgroup v2 · WireGuard · QUIC/Hysteria2 · grafos Virtuoso (RDF/SPARQL) · Radix Prefix Caching · FNV-1a + normalización $L_2$ · Chernoff-Hoeffding · suites SAST estilo CodeQL.

**Producto y fábrica BaaS/SaaS:** Next.js 16 · React 19 · Astro 5 · Apollo Supergraph (707 modelos Prisma) · 3D/WebGL (Three.js/R3F) · i18n en 9 idiomas.

**Estrategia comercial B2B2B:** precios por organización · red de afiliados con revenue share · aislamiento RBAC empresarial · alianzas institucionales (Emprendo Contigo, FuerzaParaSeguir, SocialSphere).

**Escala real:** 8 dominios en producción — el mismo stack que sostiene al sistema de salud y al de agencias.

**Green IT & eficiencia:** lzt-hub Rust 8 MB RSS + C99 zero-heap dan **−94% Joules/query** contra Python/Node ($E_\pi$ = 0.6 mJ contra 10.5 mJ, Δ −94.3%) y **−89% $CO_2$/query** · lzt-hub: 28,547 LOC, binario de 8.14 MB, 20 crates, 0 alloc en hot path.

### Vinculación con universidades · oferta de I+D colaborativo

**Cadenas de suministro · importaciones y exportaciones**

Modelos deterministas (USL + Chernoff-Hoeffding) aplicados a logística multimodal, despacho aduanal y trazabilidad. **Output esperado:** paper conjunto y piloto en una cadena real (Mérida, Veracruz, SLP) con telemetría auditable.

**Automatizaciones en microprocesadores embebidos**

Mis sistemas de compilación de bajo nivel (C99 + Rust + bare-metal) llevan LOUST a microcontroladores ARM Cortex-M, RISC-V y ESP32-S3. **Output esperado:** stack embebido con inferencia local sub-mW para telemetría industrial.

**Robótica · alianza con especialistas en mecatrónica**

Mi conocimiento de compilación de bajo nivel más el de mecatrónica da robots que corren LOUST localmente, sin cloud, con telemetría al proxy central. **Output esperado:** robot open-source que demuestra inferencia soberana y control en tiempo real, replicable en células de manufactura.

**Becas para estudiantes y coinvestigadores**

El modelo de becas de cómputo agéntico (página 4) cubre entornos aislados para estudiantes y profesores vinculados. **Cada beca es auditable** vía `/metrics` del proxy (`lzt_proxy_request_total`): la universidad ve exactamente qué se consume en su gente.

### Perspectiva de género e inclusión por diseño (impacto social real)

**1 · Democratización tecnológica para mujeres emprendedoras (+60% de participación).** En la comunidad de Emprendo Contigo en Mérida, más del 60% de los proyectos son liderados por mujeres. Autogenerarles plataformas digitales a una fracción del costo extranjero elimina la barrera económica de entrada histórica.

**2 · Equidad algorítmica (fairness by design).** Al reemplazar embeddings neuronales opacos por proyecciones matemáticas deterministas (FNV-1a, Chernoff-Hoeffding) y grafos auditables, erradicamos de raíz los sesgos demográficos que arrastran los modelos pre-entrenados comerciales.

**3 · Semillero STEM en el sureste (LZT-Developers).** Padrón abierto de talento técnico con mentoría práctica en arquitectura de sistemas, análisis SAST e IA soberana para jóvenes e ingenieras de la región sureste mexicana.

### Código Abierto Auditable (Open Source / FOSS) y Soberanía Tecnológica · panorama de mercado y por qué ganamos

**Código abierto auditable como ADN, no como marketing.** Miembro de la *Open Source Initiative*, la entidad que define qué es "open source" en el mundo; contribuidor activo en *Django Commons* (mantenimiento core del framework que sirve más de 80,000 sitios). El compromiso se traduce en código auditable, matemática pública e **independencia total**: los únicos componentes externos son los modelos de lenguaje —y son conmutables—, así que hay **cero ataduras a proveedores extranjeros** en la infraestructura.

**Panorama competitivo y por qué ganamos. Competencia IA-soberana-MX:** character.ai con $1B (a16z 2024). **LOUST ($0 levantado, ARR Y2 de $3.3M MXN, esto es $165K USD)** compite con residencia de datos en México, 0 USD en APIs de vectorización, time-to-ROI de 30 días y matemática reproducible. **Multiplicador de ahorro:** 10× por dólar invertido frente al equivalente en Python o LangChain.

**EXPLORAR PORTAFOLIO Y CV**
showcase.loust.pro

**VER COMERCIAL B2B2B**
socialspheremx.loust.pro (34+ empresas)

**CONTACTAR AL FUNDADOR**
linkedin.com/in/davidmireles ll/