

Entregable 3: Stack Técnico y Arquitectura

El Orquestador IA de Marketing: 5 fuentes vía MCP, decisiones técnicas y roadmap de closed-loop attribution

Antonio Rodríguez de Tembleque

Tutor: Cristian Betancort Niz

Máster IA Aplicada al Marketing · KSchool

FactuFicticia — SaaS B2B México



¿Por qué no Connectif?

Tres razones concretas que motivaron construir un orquestador propio en lugar de adoptar una plataforma de automatización existente.

1

Fricción de configuración

Cada fuente requería un conector específico y mapeo manual de campos. Sin acceso programático real al dato crudo, la integración era frágil y dependiente de la plataforma.

2

Caja negra parcial

Los modelos de scoring no exponen su lógica con suficiente detalle para auditar por qué un lead puntúa X. Imposible justificar decisiones comerciales ante el tribunal o el cliente.

3

Coste vs. casuística

El plan necesario para conectar GA4 + Google Ads + HubSpot a la frecuencia del negocio quedaba fuera de presupuesto. El ROI no justificaba la inversión en licencias.

i **Nuevo enfoque:** El LLM hace el rol de Connectif, con acceso programático directo a las fuentes vía MCP – sin intermediarios, sin cajas negras, sin coste de licencia.

Lo que ya funciona en producción

Resultados concretos obtenidos en las primeras dos sesiones de trabajo, con acciones ejecutadas sobre datos reales del negocio.

5

Fuentes MCP

GA4, GSC, Google Ads, DataForSEO y Skill
SEO/GEO integradas

\$6,613

MXN/mes ahorrados

4 negativas de competencia detectadas vía API
en 8 campañas

200

Líneas de código

MCP custom Google Ads (FastMCP 3.2 · stdio)

2

Sesiones de trabajo

Para montar las 5 integraciones completas

100%

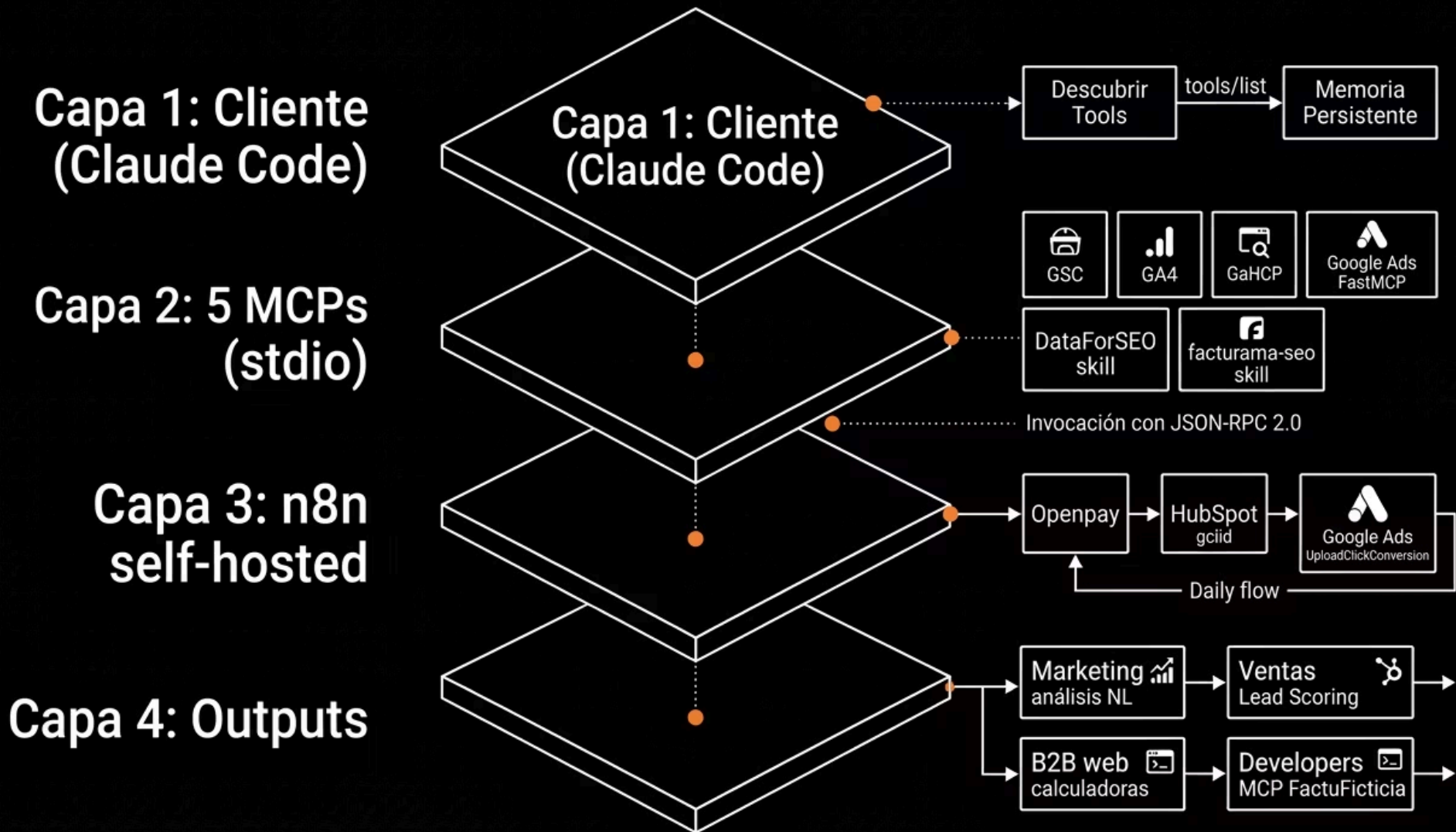
Reversible

Con un único comando API, sin impacto en
campañas activas

✓ **Acción ejecutada:** Análisis de 30 días de search terms → **Sinube, AGSA, Facturación Elite, Facturify** detectadas → Shared Negative Keyword List aplicada a 8 campañas → ahorro estimado ~\$79,356 MXN/año

Arquitectura del Orquestador IA

Cuatro capas funcionales que transforman datos crudos de múltiples fuentes en decisiones de marketing accionables, con el LLM como núcleo orquestador.



El flujo de datos es unidireccional y auditable: desde las fuentes externas hasta los outputs de negocio, pasando por un pipeline de enriquecimiento automatizado en n8n.

Las 5 fuentes integradas vía MCP

Cada fuente tiene un nivel de integración distinto: MCP comunitario, servidor custom propio o skill Python. La cobertura abarca desde analytics hasta inteligencia competitiva y contenido GEO/AEO.

Fuente	Tipo	Acceso	Cobertura principal
Google Search Console	MCP comunitario	Service Account JSON	24 tools: search_analytics, URL inspection, sitemaps
Google Analytics 4	MCP comunitario	Service Account JSON	get_ga4_data · search_schema · anti-exploración de filas
Google Ads ⚡	MCP custom propio (FastMCP 200 LOC)	OAuth desktop + refresh token	6 tools: list_accounts, account_summary, campaign_performance, top_keywords, top_search_terms, search GAQL
DataForSEO	Skill Python scripts	API key Basic Auth	keyword_ideas · serp_organic · keyword_difficulty · search_volume
facturama-seo-content	Skill propia	Triggers internos	Framework GEO/AEO 8 bloques + DataForSEO orquestado

📌 **Roadmap:** HubSpot (Fase 3) y Openpay (Fase 1) cerrarán el ciclo de closed-loop attribution, conectando la conversión real con la keyword de origen.

Servidor MCP custom: Google Ads en 200 líneas

Stack técnico

- FastMCP 3.2.4 + SDK google-ads-python 30.1.0
- Transporte: stdio (subproceso del cliente, sin puerto abierto)
- 200 LOC en `scripts/google_ads_mcp_local.py`
- Configurado en `.mcp.json` del proyecto
- 4 piezas de credencial OAuth: Developer Token · OAuth Desktop client · Refresh Token · Customer ID + MCC Login ID

6 tools expuestas al agente

- `list_accounts` – cuentas accesibles desde el MCC
- `account_summary` – KPIs agregados por rango de fechas
- `campaign_performance` – desglose por campaña con ROAS
- `top_keywords` – ordenadas por cost/clicks/conversiones/impresiones
- `top_search_terms` – search terms reales (queries efectivas)
- `search (GAQL)` – consulta SQL-like de sólo lectura

⚠ **Pitfall:** `use_proto_plus=True` no expone `DESCRIPTOR.fields` → usar `MessageToDict(row._pb)` · `cost_micros ÷ 1,000,000` para obtener el coste real en moneda.

stdio > HTTP · Custom > Oficial

El servidor oficial de Google Ads falló con `MCP error -32000: Connection closed`. La decisión de reescribirlo en stdio fue técnica, no estética.

❓ Servidor oficial (HTTP + OAuth 2.1)

- Streamable HTTP en puerto 8080 expuesto
- OAuth 2.1 con Dynamic Client Registration
- Cliente debe completar OAuth dance completo
- Falla: cliente no soporta DCR maduro
- Requiere OAuth Web client con redirect URI configurado

✓ MCP custom (stdio)

- Compatibilidad universal: todo cliente soporta stdio
- Sin red expuesta: corre como subprocesso local
- Un único refresh token, sin segundo OAuth dance
- Control del catálogo: 6 tools enfocadas, no 40+
- OAuth Desktop + InstalledAppFlow simplificado

📌 **Lección general:** stdio > HTTP para MCPs locales · Custom > oficial cuando el oficial falla por transporte · Skill > MCP cuando integras una metodología, no una fuente de datos.

Skill facturama-seo-content: Framework GEO/AEO

Adapta los hallazgos de Princeton 2024 (Aggarwal et al.) al contexto fiscal mexicano. Ocho bloques metodológicos que generan briefs editoriales con señales GEO específicas para dominios .mx.

01 Intent fit

02 E-E-A-T fiscal

03 Estructura snippets

04 GEO signals

UMA, Art. 87 LFT, Art. 96 LISR

05 Schema JSON-LD

06 Cluster + internal linking

07 Anti-AI heuristics

08 Performance técnica

CWV, INP

Keyword research

Extraer términos GEO relevantes

Keyword difficulty

Evaluar competencia y prioridad

SERP analysis

Analizar resultados y entidades

Brief editorial

Outline, schema y checklist GEO/AEO

El pipeline orquesta DataForSEO y la lógica propia para producir briefs listos para redacción, con señales GEO específicas para el mercado fiscal mexicano.

Cerrando el Bucle de Atribución

⊗ **Constraint:** trial 30 días → primera compra Openpay al día 31-32, fuera de la ventana default de Google Ads → las compras reales **NO se atribuyen** con la configuración estándar.

1

① Openpay

Cargo del día vía REST API: email, amount, date. n8n pollea diariamente.

2

② HubSpot

Lookup gclid por email. HubSpot Starter guarda el gclid del clic al registro.

3

③ Google Ads

UploadClickConversion: gclid + acción "Compra Openpay offline" + valor dinámico.

4

④ Config complementaria

lookback_window → 90 días · GA4 User-ID · BigQuery export activo.

5

⑤ Resultado en 6 semanas

ROAS real por keyword · LTV por canal · validación con datos económicos reales.

5 Pitfalls Reales Documentados

Aportación técnica: problemas reales encontrados integrando APIs Google + Python, con soluciones reproducibles.

Rutas Windows con espacios

Síntoma: pipx falla con username "El minador" en la ruta.

Solución: `python -m pip` directo en Python embebido, sin pipx.

CRLF en tokens OAuth

Síntoma: `invalid_grant: Bad Request` al refrescar token.

Solución: Verificar bytes raw del token; nunca copiar desde editores que añaden EOL.

`use_proto_plus=True`

Síntoma: No expone `DESCRIPTOR.fields` en respuestas de Google Ads API.

Solución: `MessageToDict(row._pb, preserving_proto_field_name=True)`.

Ventana conversión 30 días

Síntoma: Compras PPD no se atribuyen al canal de origen.

Solución: Extender a 90 días + Offline Conversion Import activo.

Tablas grandes en LLM

Síntoma: Saturan contexto y degradan calidad de respuesta.

Solución: Procesar con scripts Python; presentar sólo agregados y top N.