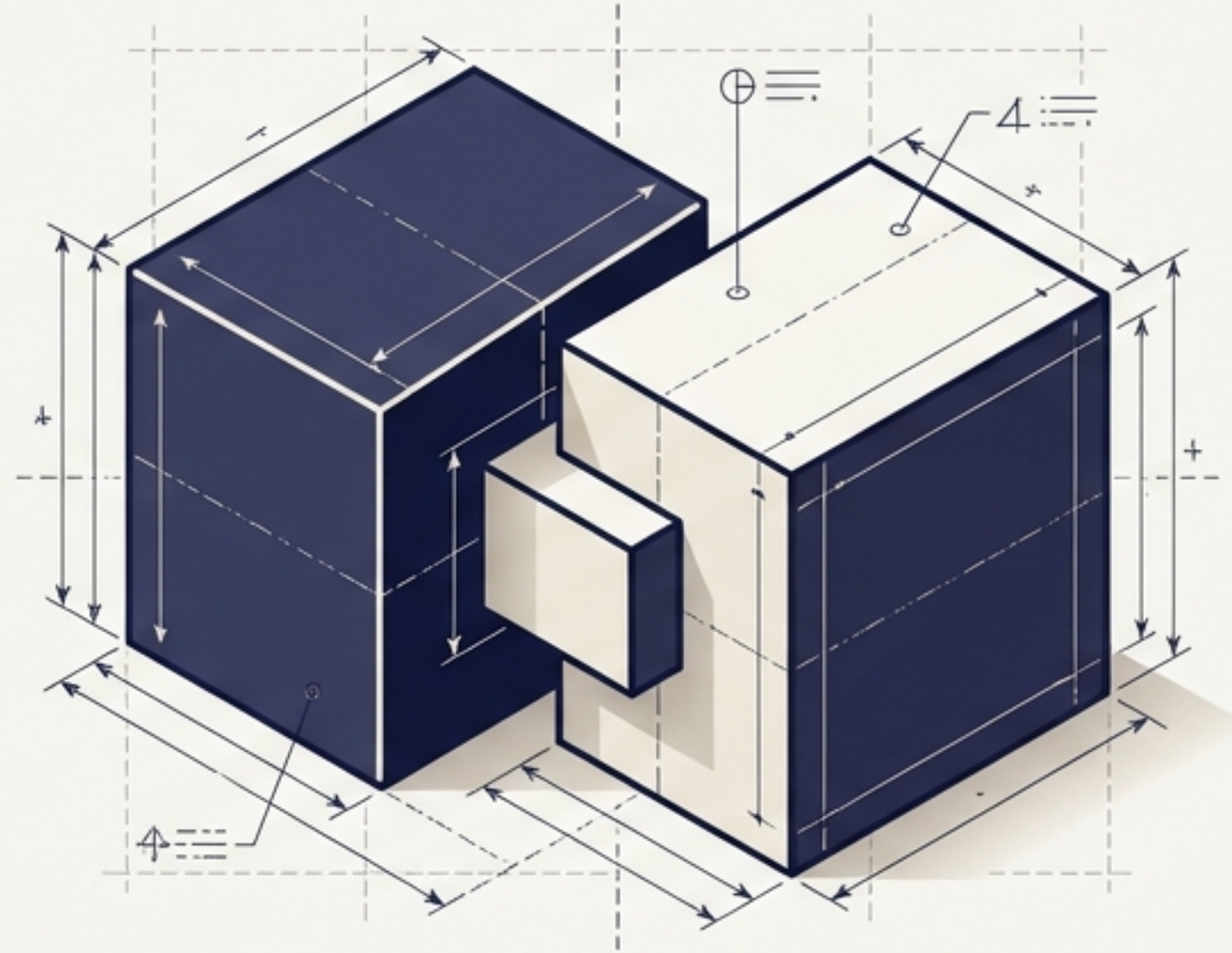




El Apretón de Manos Inteligente

Arquitectura MCP para agentes de IA que no adivinan tu negocio.

Basado en la implementación de producción de PeopleworksGPT (.NET 9, C#).



El problema de la “Amnesia” de la IA

Hay algo que nadie quiere decir en voz alta sobre los agentes de IA: cada agente que desplégas empieza desde cero absoluto en cada sesión.

Sabe SQL y estadística. No sabe que a los empleados les llamás “asociados”.

No sabe que VentasBrutas ya excluye las devoluciones.

No sabe que el CFO siempre exige las cifras en USD y MXN.

El **AI adivina**. Y cada suposición incorrecta es una respuesta errónea entregada con plena confianza.

El punto ciego del MCP

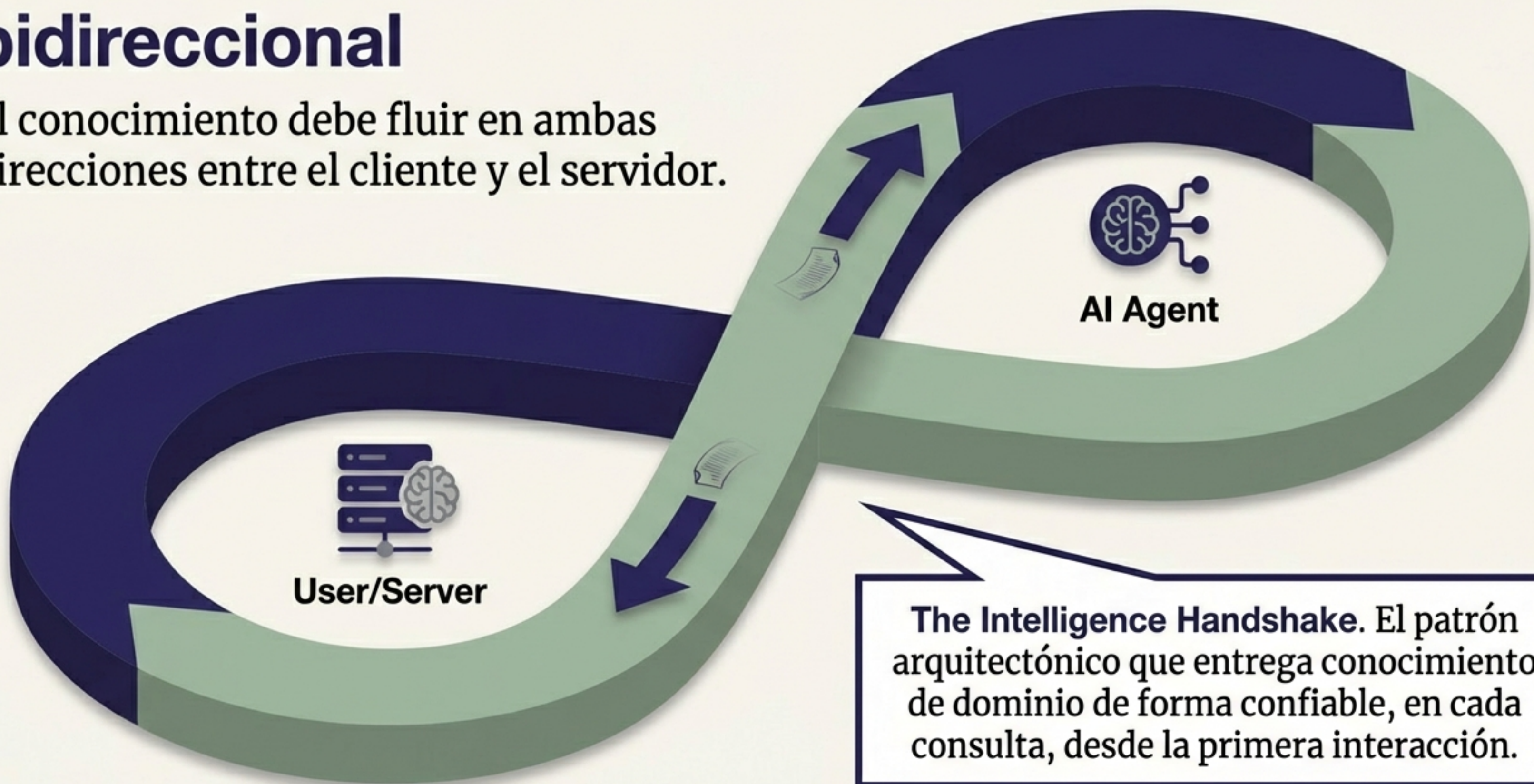
MCP es un estándar poderoso, pero casi todas las implementaciones comparten el mismo defecto arquitectónico: el contexto solo fluye en una dirección.



El usuario hace una pregunta aislada. El AI responde sin contexto del dominio. Faltan reglas de moneda, filtros de registros activos y nombres correctos de columnas.

La revelación: El contexto bidireccional

El conocimiento debe fluir en ambas direcciones entre el cliente y el servidor.



Los dos canales de comunicación

Canal 1: Servidor → Cliente AI

El servidor expone su inteligencia acumulada al inicio de la sesión.

Transmite: Reglas de negocio, significado de tablas, historial de consultas.

Método: `GetConversationContext()`

Canal 2: Cliente AI → Servidor

El cliente AI pasa su foco conversacional actual para generar análisis, no respuestas aisladas.

Transmite: Hilo conversacional, foco de análisis (ej. 'comparativa Q4').

Método: `ExecuteQueryAsync(clientContext)`

Pilar 1: El primador de sesión

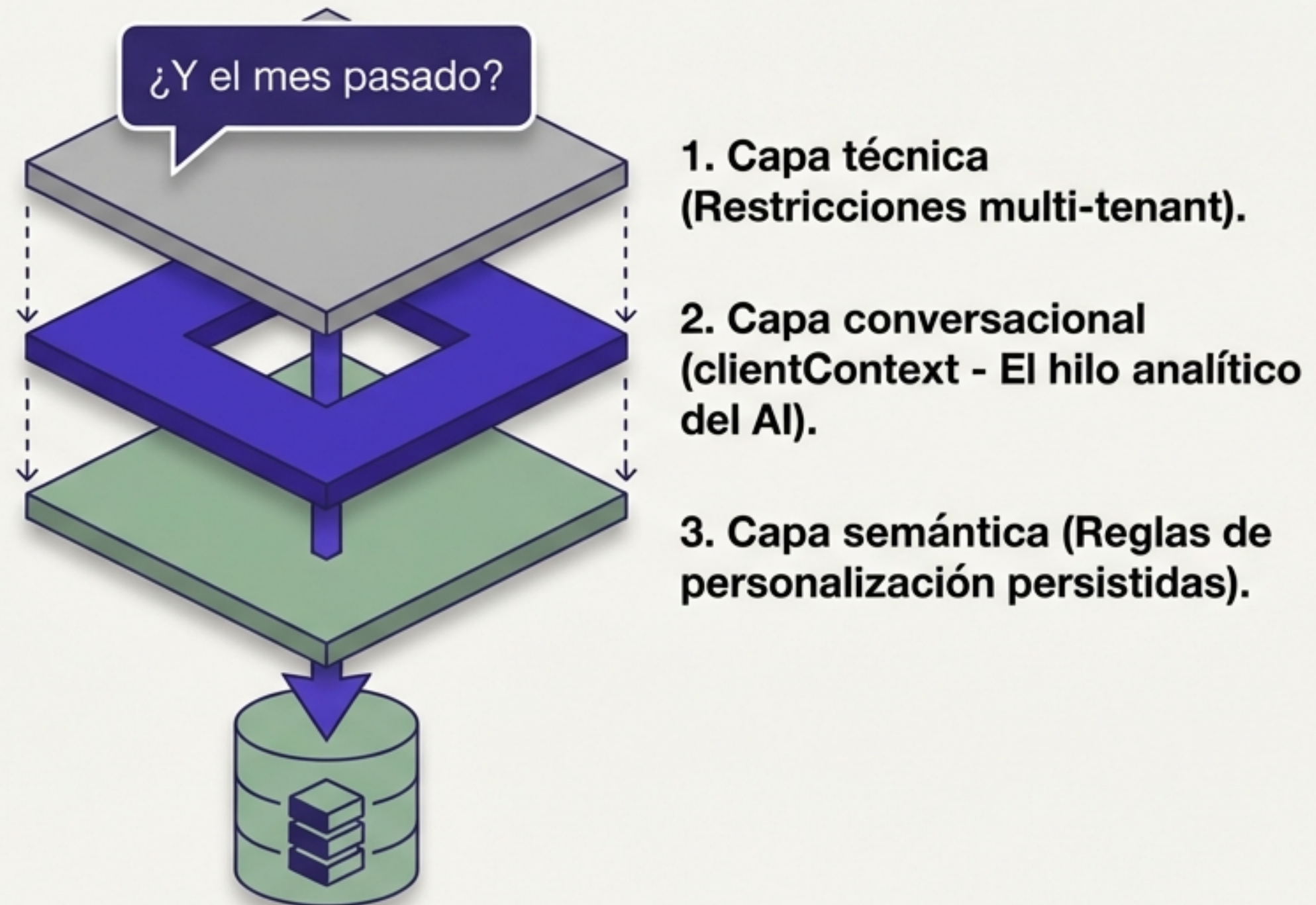
El servidor agrega todo el conocimiento del dominio en una sola llamada, generando un **client_priming_guide** que el AI referencia durante toda la sesión.



```
[  
  "client_priming_guide":  
  {  
    "Reglas de Negocio": "Usar 'asociados' en vez  
de 'empleados'. Filtrar por Status = 'A'.",  
    "Personalización": "Mostrar valores en USD y  
MXN por separado.",  
    "Historial": "'Top 10 asociados en Q4'  
(Referencia de contexto)."  
  }  
]
```

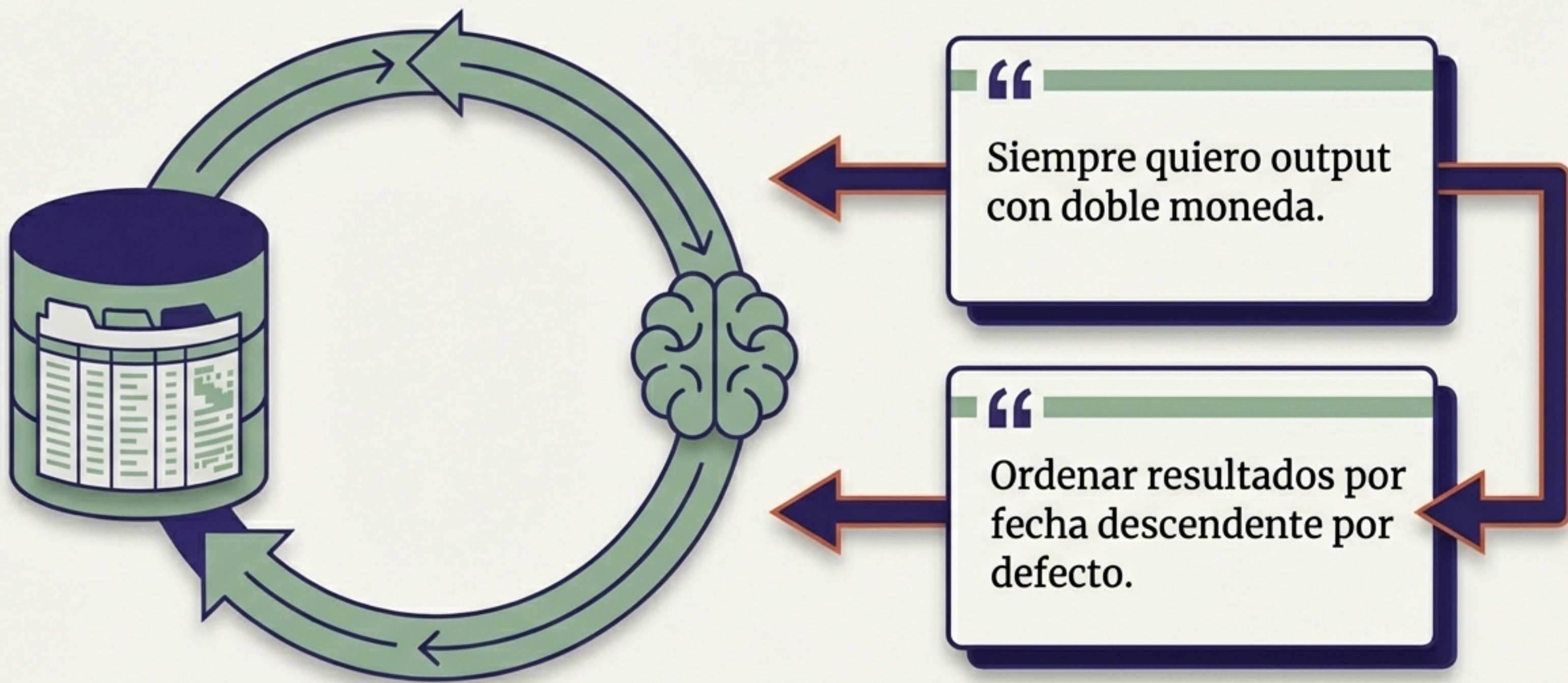
Pilar 2: El foco conversacional

Sin este canal, el servidor responde a la pregunta aislada. Con él, el AI comparte lo que sabe sobre la conversación actual.



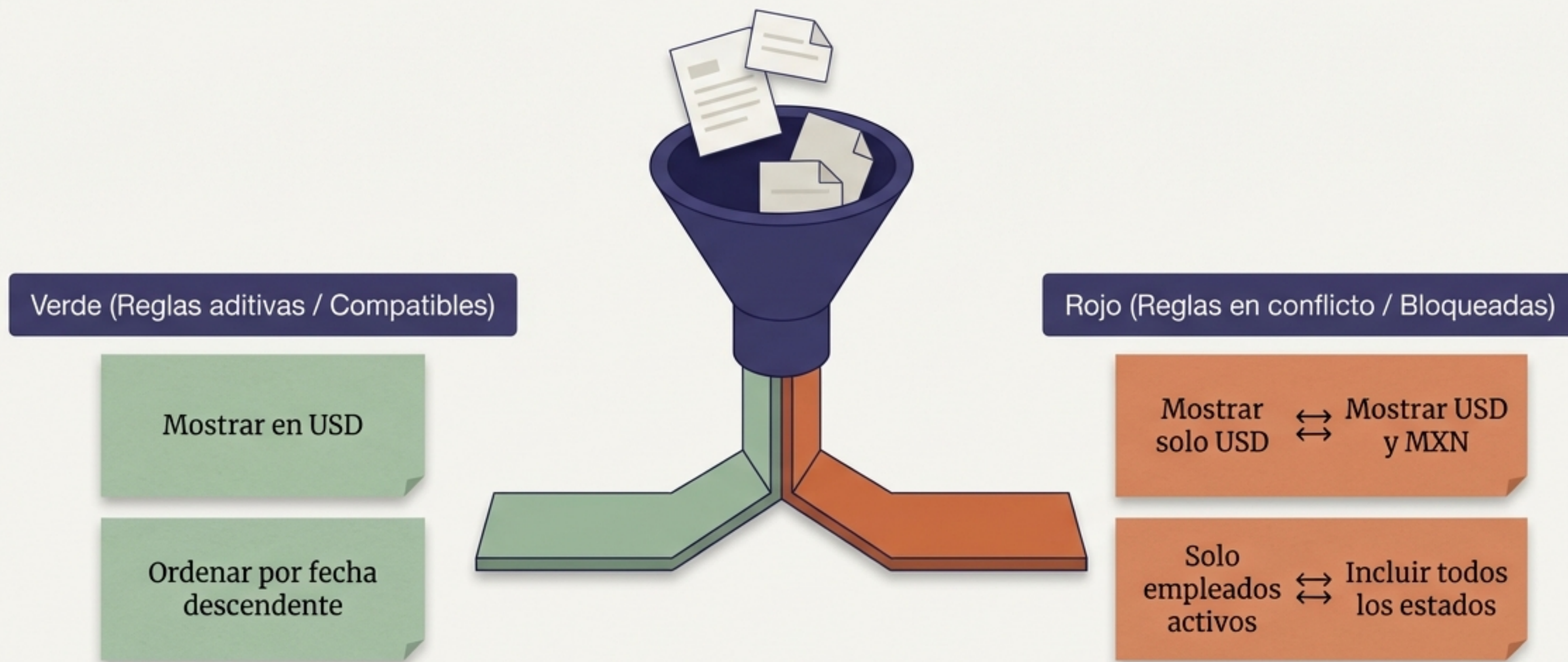
Pilar 3: Inteligencia persistente del usuario

Memoria que sobrevive a las sesiones. El AI aprende una sola vez las preferencias del usuario y las inyecta automáticamente en cada consulta futura.



El desafío de la persistencia: Detección de conflictos

Antes de guardar una nueva regla, el sistema usa IA para verificar si contradice el contexto existente.



El disparador determinístico

Escribir instrucciones en la descripción del tool es probabilístico. Necesitamos comportamiento determinístico.

La Solución: Incrustar las instrucciones de 'priming' en la respuesta de autenticación obligatoria.

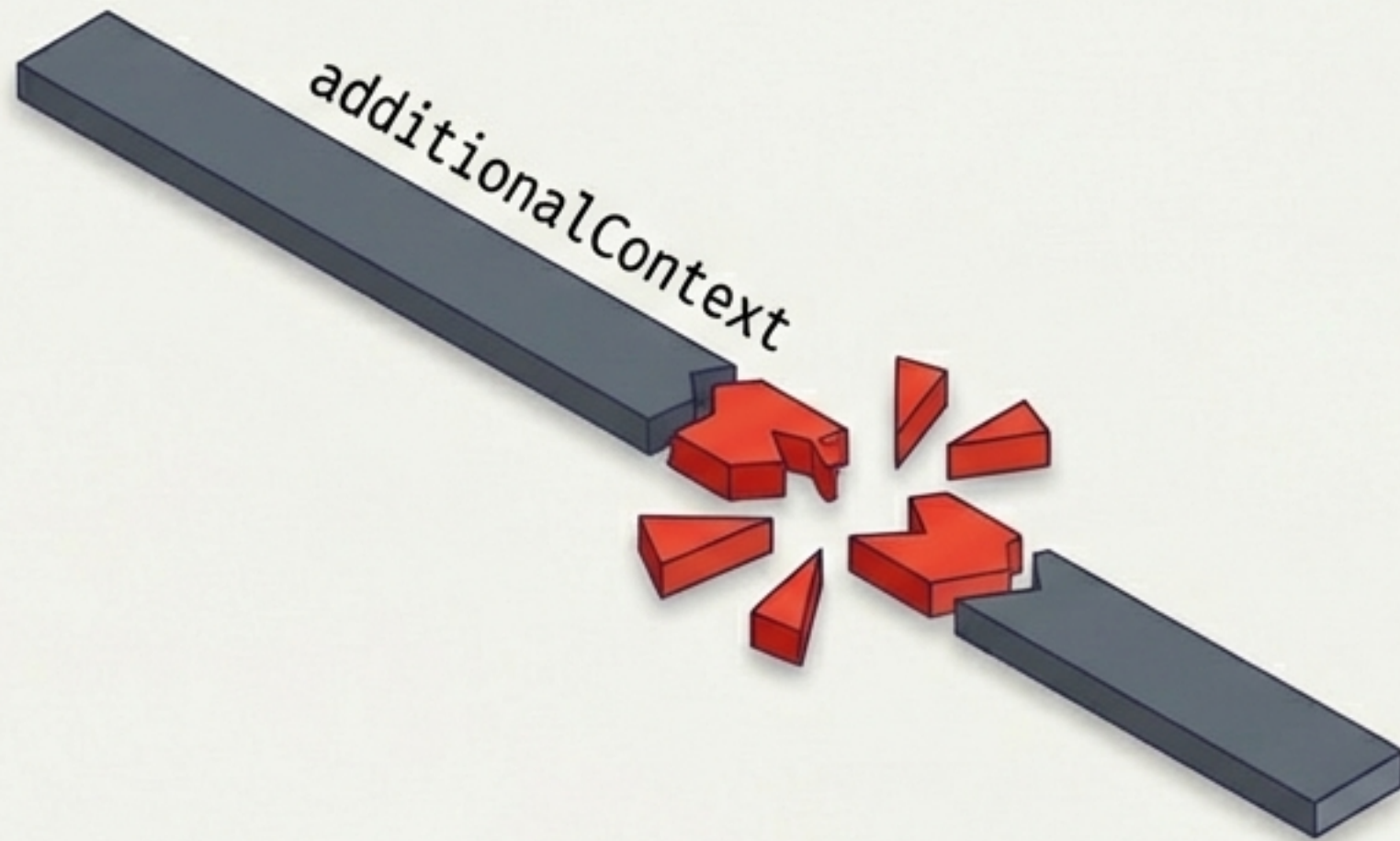
```
return JsonSerializer.Serialize(new {  
    success = true,  
    session_token = sessionToken,  
    next_step = new {  
        action = 'prime_session_context',  
        instructions = '1. Llamar a GetConversationContext...'  
    }  
});
```



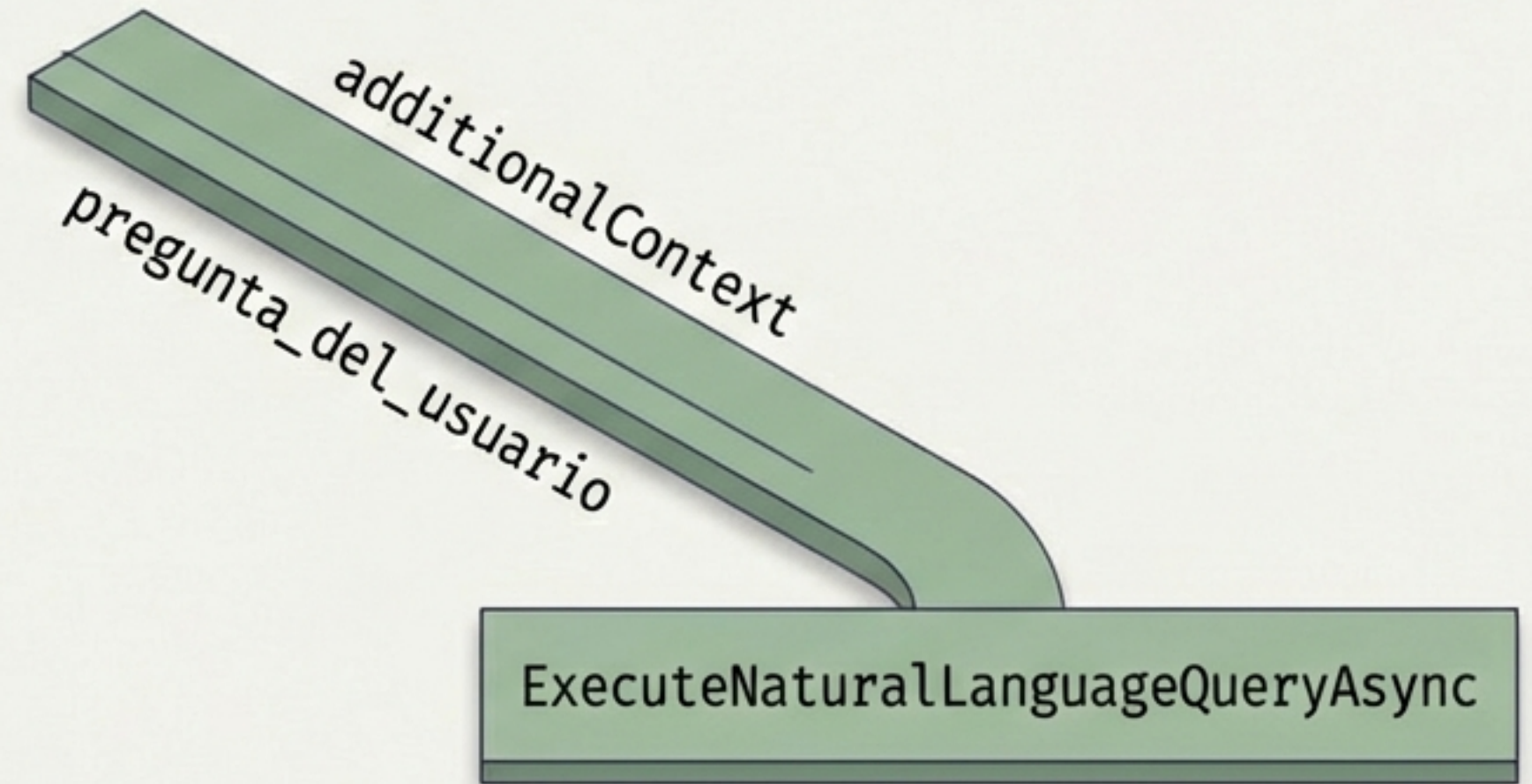
Los LLMs tratan los 'tool results' en vivo con alta fidelidad. El AI debe procesar el bloque next_step porque está en su context window activo, forzándolo a leer la guía de inicio antes de ejecutar consultas.

Alerta arquitectónica: El bug silencioso

Un error invisible en las pruebas que destruye el contexto. El método de servicio acepta los parámetros, pero los descarta silenciosamente antes de llamar al modelo.



El Síntoma: Las consultas funcionan, pero son genéricas. El AI nunca recibió el contexto que cuidadosamente construiste.



La Solución: Asegurar que `additionalContext` se concatene con la pregunta del usuario antes de enviarla.

La prueba empírica: Antes vs. Después

Consulta: “Mostrame los mejores vendedores del trimestre pasado”

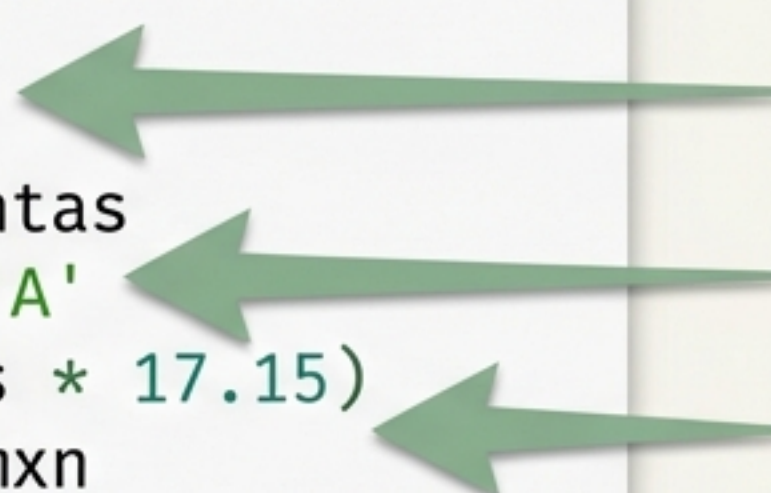
Sin Handshake - Genérico

```
SELECT  
FROM empleados
```

- ✗ Error: tabla incorrecta
- ✗ Falta filtro de status 'A'.
- ✗ Falta control de devoluciones.

Con Handshake - Domain-Aware

```
SELECT  
  FROM Asociados a  
  JOIN RegistrosVentas  
WHERE a.Status = 'A'  
SUM(v.VentasBrutas * 17.15)  
AS total_ventas_mxn
```



- ✓ Tablas correctas
- ✓ Regla aplicada
- ✓ Regla de doble moneda aplicada

Un patrón universal más allá de SQL

Cualquier servidor MCP que opere en un dominio específico se beneficia de esta arquitectura.



Agentes CRM / ERP

Mapear que “oportunidades cerradas” en Salesforce equivalen al “status 7” en el ERP legado.



Análisis Financiero

Enseñar al agente que el EBITDA de esta empresa incluye depreciación de equipos, por definición del CFO.



DevOps

Instruir que “prod-db-04” es réplica de reportes, bloqueando consultas DELETE.



MCP

La inteligencia es arquitectura

Los agentes de IA son tan inteligentes como el contexto que llevan. La pregunta no es si el AI es suficientemente capaz, sino si construiste la arquitectura para que esa inteligencia cuente.

No es una configuración. No es un prompt. Es un protocolo.