# El Límite Cognitivo: Por qué la IA Local Falla en la Gobernanza Documental

Compendio | Vol. 1

**mercedev.es** — 2026-05-08 | Fase Épica 2 - Fase 3 (Orquestación de Contenidos)

## El Desafío (Síntoma)

La Fase 3 del Roadmap de IA tenía un objetivo claro: descentralizar la gobernanza documental construyendo agentes autónomos locales ( `merci-librarian` y `merci-ssot` ) capaces de redactar cuadernillos y mantener sincronizado el Roadmap sin depender de APIs de terceros (Gemini).

Durante la implementación empírica, los modelos locales probados (Phi3 3.8B, Llama 3 8B, Qwen2.5 7B y Qwen3.5 9B) fracasaron sistemáticamente. Los agentes sufrían de *Context Window Stuffing* (amnesia por exceso de contexto), *Recency Bias* (priorizar la última línea leída ignorando el prompt principal) e incapacidad para seguir el *Zero-Shot formatting* estricto requerido por los YAML Frontmatters, destruyendo a menudo los archivos de salida con texto conversacional no deseado.

## La Maniobra (Lógica)

Para intentar domar a los modelos locales, se desplegaron escudos DevSecOps de alto nivel: 1. **Inyección de Plantillas (One-Shot Prompting):** Proveer el molde exacto para frenar la creatividad del modelo. 2. **Filtrado Semántico RAG:** Reducir la inyección del contexto a menos de 3000 caracteres extrayendo solo las entradas relevantes basadas en palabras clave. 3. **Resource Budgeting y Trazabilidad:** Forzar el límite de contexto ( `4096` ), matar la temperatura ( `0.0` ), ampliar los *timeouts* ( `600s` ) y utilizar herramientas de red de bajo nivel ( `tcpdump` ) para depurar el payload.

A pesar de estas contenciones de infraestructura, el análisis de red reveló la carencia cognitiva estructural de estos modelos: gastaban el 99.9% de su cuota de *tokens* en monólogos internos (Reasoning) intentando comprender la instrucción de formato, agotando la memoria antes de poder redactar el documento final.

## El Aprendizaje / Deuda Técnica

Se concluye que **la Gobernanza Documental en texto plano y Markdown estricto es territorio exclusivo de los Modelos de Frontera (Cloud).**

Los Small Language Models (SLMs) locales en hardware de consumo (<14B parámetros) son excelentes para tareas de sintaxis corta, revisión de código o detección de errores (como el Agente Auditor), pero carecen de la atención sostenida necesaria para la reescritura masiva de documentos y el cumplimiento de formatos estructurales complejos. El ecosistema asume la deuda de depender de APIs como Gemini Flash para los agentes de la Fase 3, consolidando un *Hybrid Stack* donde la nube absorbe la carga semántica y el local la sintaxis.