



Debrief — google/gemma-4-26b-a4b-it

- Instanser: 9
 - Tokens inn: 20,694
 - Tokens ut: 8,222
 - Kostnad: \$0.0045
 - Kjørt: 2026-04-14 16:15:27.833237+00:00
-

Sverm-debrief: Implementering av KI-sverm i helseorganisasjon

Konsensus

Svermen er enstemmig på at **Copilot Studio/Power Automate er utilstrekkelig** for de forespurte use-casene. For å skalere til 100 agenter og håndtere kompleksiteten i Case 1, må arkitekturen flyttes til en **Python-basert orkestrering (LangGraph eller Semantic Kernel)** deployert på **Azure Container Apps/Functions**.

Nøkkelpunkter for implementering:

- **Arkitektur:** En hybrid modell som bruker Azure OpenAI (AOAI) for selve intelligensen, men flytter orkestreringen ut av SaaS-begrensningene for å sikre kontroll over state management og concurrency.
- **Sikkerhet:** Data må forblir i din Azure-tenant. Bruk **RAG via Azure AI Search** med **Entra ID-basert security trimming** (agenten arver brukerens tilgangsnivå) og **Managed Identities** for å eliminere behovet for lagrede credentials.
- **Skalering (Case 1):** Bruk en **hierarkisk sverm-modell** (Orchestrator → Domain Experts → Specialist Agents) for å unngå informasjonskaos.
- **Skalering (Case 2):** Implementer en **asynkron kø-modell** via **Azure Service Bus** for distribuert prosessering.
- **Kostnad:** Bruk en "Tiered Model"-strategi: GPT-4o for orkestrering/syntese, og GPT-4o-mini for de 100 arbeider-agentene.



Dissens

Det var ingen fundamental uenighet om teknologistakk, men ulike vektlegginger av **risiko vs. makt**:

- Noen instanser fokuserte på den **juridiske/regulatoriske kontrollen** (sikre at agenter ikke går i “feedback loops” eller hallusinerer i helsedata).
- Andre la vekt på den **strategiske makten** ved å eie orkestreringslogikken selv, fremfor å være låst til Microsofts “black box”-oppdateringer.
- Det var en nyanse i hvordan syntese bør skje: Noen foreslo en enkel Map-Reduce, mens andre krevde en Conflict Matrix for å eksponere uenighet mellom agenter fremfor å tvinge frem falsk konsensus.

Blindsoner avdekket

Svermen identifiserte kritiske risikoer som en enkelt analyse ville oversett:

- **Semantic Drift/Entropy**: Risikoen for at 100 agenter i en sverm skaper en “ekkokammer-effekt” eller går inn i destruktive feedback-loops.
- **Token-bleed**: Den økonomiske risikoen ved at parallelle agenter i en uregulert sverm kan tømme budsjetter på minutter uten en sentralisert Token Proxy/Budgeter.
- **Complexity-to-Value Ratio**: Advarsel om at sverm-arkitektur er en overkill for lineære oppgaver; kompleksiteten i vedlikehold kan utradere ROI hvis oppgaven ikke er genuint multidimensjonal.

Anbefalinger

1. **Etabler en Azure Landing Zone**: Sett opp isolerte miljøer med Managed Identities og Azure AI Search som er ferdig konfigurert for helse-compliance.
2. **PoC (Fase 1 - Case 2)**: Start med distribuert batch-prosesserings-kø (Service Bus → Python → GPT-4o-mini). Dette er lavest risiko og gir raskest ROI.
3. **Utvikle en “Persona Factory”**: Bygg en metadata-drevet motor i Python som injiserer unike system-prompts og tilgangsnivåer til agenter, fremfor å hardkode prompts.
4. **Implementer Observabilitet fra dag én**: Integrer Azure Monitor/Application Insights med en `CorrelationID` som sporer en case gjennom hele svermlivssyklusen for å overvåke både kostnad og feilrate.