



id: SI_004

model: anthropic/claude-sonnet-4.6

seeds:

04_risiko: sårbarhet

03_relasjon: tillit

05_handling: aksepter

01_perspektiv: operasjonell

02_tidhorisont: langsiktig

Instans SI_004

Seeds: operasjonell | langsiktig | tillit | sårbarhet | aksepter

Kjerneperspektiv: Aksepter sårbarhetene — bygg tillit operasjonelt

For en 200-ansatts helseorganisasjon med streng sikkerhetskontroll er det kritiske spørsmålet ikke hvilken swarm-arkitektur, men hvilken sårbarhet du aksepterer å eie.

Arkitektur-valg: Azure OpenAI er eneste reelle svar

Copilot Studio + Power Automate er **ikke** en swarm-plattform. Det er en workflow-automatisering med LLM-wrapper. For 100 parallelle agenter med differensiert kontekst trenger du:

Azure OpenAI + Azure Durable Functions (cloud-only, ingen Kubernetes):

- Durable Functions håndterer fan-out/fan-in uten infrastrukturarbeid
- Data forlater aldri Microsoft-tenanten (kritisk for helse-data)
- Entra ID-integrasjon er native

Sårbarhet du må akseptere: Azure OpenAI har lavere modell-diversitet enn Anthropic/OpenAI direkte. GPT-4o er din arbeidsmodell. Ingen Claude, ingen Gemini — med mindre du eksplisitt godtar data-residency-risiko.



Agent-spesialisering: Rollen er kontekst, ikke prompt

```
# Pseudo-arkitektur: rolle-differensiert kontekst-injection
agent_roles = {
    "regulatory": {"system": REGULATORY_PROMPT, "context_filter": ["lovverk",
"GDPR", "Helsepersonelloven"]},
    "operational": {"system": OPS_PROMPT, "context_filter": ["ressurser",
"kapasitet", "SLA"]},
    "risk": {"system": RISK_PROMPT, "context_filter": ["hendelser",
"avvik", "ROS-analyser"]}
}
# SharePoint-data hentes via Graph API med brukertokens – aldri service
principal med full tilgang
```

Aggregering: Én dedikert synteseagent leser alle outputs og produserer strukturert JSON-rapport. **Ikke konsensus — hierarki.**

Langsiktig tillit krever observerbaritet fra dag én

Azure Monitor + Application Insights må konfigureres før produksjon:

- Token-forbruk per agent-rolle (cost attribution)
- Feil-rate per case-kompleksitet
- Latency-distribusjon (batch på timer er OK, men outliers avslører designfeil)

Failure mode å akseptere: Swarm gir ikke bedre svar enn én god agent på veldefinerte problemer. Den gir verdi kun der perspektiv-bredde er genuint nødvendig.

Realistisk timeline

- **Måned 1-2:** Azure OpenAI-tilgang, Entra-integrasjon, Graph API-tilkobling, 3-agent pilot
- **Måned 3-4:** Durable Functions fan-out for Case 2, observerabilitet på plass
- **Måned 5-6:** Case 1 med 10→100 agenter, syntese-lag, kost-validering

Neste steg

1. **Bestill Azure OpenAI-tilgang med norsk data-residency** (Norway East) — dette er uken-ett-oppgaven



2. **Bygg én Durable Functions-orchestrator** for 5 parallelle agenter på et ikke-sensitivt case — validér fan-out/fan-in før du skalerer
3. **Definer “tillit-terskel”**: hvilken menneskelig review kreves før swarm-output brukes i klinisk beslutning? Dokumentér dette før første produksjonskjøring