



id: SI\_002

model: anthropic/claude-sonnet-4.6

seeds:

04\_risiko: sårbarhet

03\_relasjon: makt

05\_handling: implementer

01\_perspektiv: strategisk

02\_tidhorisont: langsiktig

## Instans SI\_002

**Seeds:** strategisk | langsiktig | makt | sårbarhet | implementer

### Maktperspektivet ingen snakker høyt om

Det strategisk kritiske valget er ikke teknisk — det er **hvem som eier orkestreringslaget**. I helsevesen med 200 ansatte og streng datasensitivitet er dette et maktspørsmål med reelle konsekvenser.

### Arkitektur-valg: Sårbarhetskart

**Copilot Studio + Power Automate** gir rask implementering, men skaper strukturell sårbarhet: Microsoft kontrollerer agentlogikken, prismodellen og databehandlingsavtalene. For helsevesen betyr dette at du ikke vet nøyaktig hvor inferens kjører — et compliance-problem, ikke bare et teknisk en.

**Azure OpenAI + Semantic Kernel** (Microsoft-rammeverk for multi-agent) er det strategisk forsvarlige valget:

- Data forblir i norsk/EU Azure-region ( norweast )
- Entra ID-autentisering gjennom hele kjeden
- Du eier orkestreringskoden

```
# Semantic Kernel – agent-spesialisering (pseudo)
kernel = Kernel()
kernel.add_service(AzureOpenAIChatCompletion(
```



```
        deployment="gpt-4o", endpoint=AZURE_ENDPOINT
    ))
    regulatory_agent = ChatCompletionAgent(
        kernel=kernel,
        name="RegulatoryReviewer",
        instructions="Du vurderer kun GDPR/Normen-compliance...",
    )
    # Hver agent: ulik system-prompt + ulik RAG-kontekst fra SharePoint
```

## Case 1: 100 agenter uten kaos

Rolle-differensiering krever **hierarkisk orkestrering**, ikke flat sverm. Implementer tre lag:

1. **Taktisk lag** (~80 agenter): domenespesifikke reviewere med smal kontekst
2. **Synteselag** (~15 agenter): krysser domenefunn, identifiserer spenninger
3. **Beslutningslag** (1 orkestratoragent): produserer strukturert output

Bruk Azure Service Bus som kø — dette gir batch-prosessering og full auditlogg uten ekstern dataoverføring.

## Case 2: Inbox-modell

```
SharePoint/Dataverse → Azure Service Bus → Agent Pool
                        ↓
                        Complexity scorer (GPT-3.5)
                        → Enkel case: 1 agent
                        → Kompleks case: 3-agent subsverm
```

Complexity-scoring er undervurdert: tildel ressurser proporsjonalt, ikke flatt.

## Langsiktig sårbarhet du må adressere nå

Azure AI Foundry (tidligere AI Studio) er den strategiske plattformen Microsoft bygger rundt. Bygg mot den — ikke Power Automate-flows som låser deg i lavkode-paradigmet når behovene skalerer.



---

## Neste steg

1. **Uke 1-2:** Etabler Azure AI Foundry-workspace i `norwayeast` med Entra ID-integrert tilgangskontroll — dette er fundamentet alt annet hviler på
2. **Uke 3-6:** Implementer Case 2 først (inbox-modell er enklere å validere) med 5-10 agenter på reelle lavrisiko-cases
3. **Uke 7-12:** Evaluer Semantic Kernel vs. AutoGen for Case 1-orkestrering — test hierarkisk syntese på ett reelt komplekst problem før du skalerer til 100 agenter