

KS Digital · webinar

KI-arkitektur for kommunen

Lån hjernen, ei hukommelsen

Referansearkitektur under realisering

Rune Ødegård

Rådgiver | Plan og bygg | Kristiansand Kommune

25+ år i PwC, Tieto, Linpro, DNV, Evry, Atea og Capgemini

Tidligere Principal Solution Architect · RTE for 130 utviklere i globalt SAFe-oppsett



Kristiansand
kommune



Kan i prinsippet spres fritt med tydelig kildehenvisning til Kristiansand kommune. Deles ikke utenfor offentlig/kommunal sektor uten forespørsel til Kristiansand kommune.

Utgangspunktet

«What is the smallest system that actually solves this problem, including its foreseeable operational requirements?»

Complexity where the problem demands it.
Simplicity everywhere else.



Kjente byggeklosser | med vilje

Kompleksiteten holdes på komponenter kommunal IT allerede kjenner og drifter.

Kjent fra kommunal drift

- Virtualisering og virtuelle maskiner
- Containere (Docker)
- SQL-databaser (PostgreSQL eller MS SQL)
- API-er, nøkler og nettverkssoner
- Versjonskontroll (Git)
- Rapportering (Power BI)

Nytt å lære

- Inferensmotorer og modell-livssyklus
- LLM-gateway: ruting mellom modeller
- MCP som kontrakt mot agenter
- Datatransformasjon som kode (dbt)
- GPU-drift

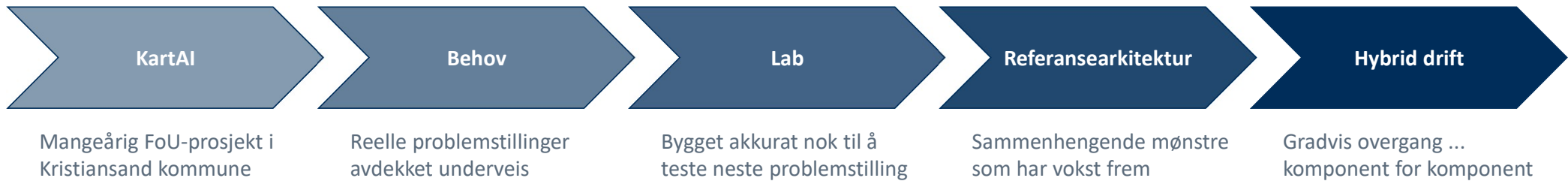
Vår vurdering:

ikke diametralt annerledes enn det kommunene allerede drifter.

Det nye er sammensetningen og anvendelsen ... ikke byggeklossene.

Vokst ut av FoU | ikke tegnet på forhånd

Arkitekturen har modnet gjennom behov avdekket i det mangeårige FoU-prosjektet KartAI.



Åpent og etterprøvbart

Kristiansand kommune har som mål å publisere resultatene åpent på GitHub.

Forskningsresultatene fra KartAI er allerede publisert der, og forklart på labs.geokrs.no. Her linkes det også til github repoene våre.

Derfor er dette ikke en stor målarkitektur der vi først tegnet og så forsøkte å realisere.

Det meste er bygget, testet eller konkretisert gjennom reelle behov.

Målet: kortere vei til riktig fagkunnskap

Teknologien er et virkemiddel. Gevinsten skal komme i saksbehandlingen.

For kommunen

- Raskere tilgang til relevant fagkunnskap
- Mindre tid på å lete etter rutiner, regelverk og dokumentasjon
- Frigjort kapasitet til faglige vurderinger
- Mer konsistente svar og arbeidsprosesser
- Organisatorisk læring og kunnskapsdeling



Konkret i Plan og bygg

- Kortere vei til riktig informasjon
- Mindre avhengighet av enkeltpersoner
- Mer enhetlig praksis på tvers av seksjoner
- Bedre utnyttelse av eksisterende kompetanse
- Håndtere økt saksmengde uten tilsvarende økning i ressursbruk

Første fase: riktig plattform og intern støtte og opplæring til medarbeidere.

Hva dette er | og hva det ikke er

Si det tidlig, så vi diskuterer det som faktisk er bygget.

Dette er

- En fungerende referanseimplementering for 1–5 brukere
- Validering av arkitekturmønstre, grensesnitt og leverandøruavhengig sammensetning
- Lokal, produksjonsnær kapasitet i et atskilt labmiljø som gradvis beveger seg fra hybrid til produksjon

Dette er ikke

- En skalert kommunal plattform for alle saksbehandlere
- Dokumentasjon på kapasitet, SLA eller sikkerhet i stor skala
- En sikkerhetsrevisjon eller lasttest

Må på plass før bred produksjon

- Governance
- Identitet/SSO og delegert autorisasjon
- Tilgangskontroll mot data og dataseparasjon
- Sporbarhet og revisjon
- FDV og operasjonell skalering

Bevisst prioritering:

bygg akkurat nok til å teste neste problemstilling ... og design slik at kontrollpunktene ikke låses ute.

KI stopper sjelden på teknologien

Den stopper på to organisatoriske spørsmål.

1 Kunnskapsforvaltning

- Hvilke dokumenter er gyldige?
- Hvem eier dem, og hvor ligger de?
- Hvem oppdaterer dem?
- Hva er fasit?

«Uten svar på dette blir KI et raskere søk i feil kilder.»

2 Kostnadsmodell for KI

- Hvem betaler for tokens, kreditter og kapasitet?
- Hvordan fordeles kostnaden mellom enheter?
- Hva skjer når bruken vokser?

«Når dette er uavklart, stopper realiseringen, selv med lisenser og teknologi på plass.»

Begge er spørsmål en nasjonal ramme bør adressere ... ikke bare valg av teknologi og leverandør.

KI-forbruk følger en annen økonomi enn lisenser

Kostnaden flytter seg fra «per bruker» til «per bruk».

Her er det et kommunikasjonsgap mellom teknologene og direktørene

Klassisk lisensmodell



- Pris per bruker eller enhet
- Kostnaden er i hovedsak kjent når avtalen inngås
- Økt bruk endrer i liten grad prisen

Relativt "kjent" kostnad pr. ansatt => lett å budsjetter

Forbruksbasert KI



- Variabel kostnad per kall, token, modell og kontekstlengde
- Volumet vokser med bruk, agenter og automasjon
- Krever fordeling av kostnad mellom enheter
- I tillegg: kreditter og kapasitet – ikke alt er tokens

HELT ukjent kostnad pr. ansatt/seksjon => VANSKELIG Å BUDSJETTER GODT

"Hvor langt er et tau?"

Hva det betyr for arkitekturen

Én vei inn

All modelltrafikk gjennom én gateway
(lastbalansert ved behov).

Nøkler og budsjetter per bruker eller enhet
gjør måling og fordeling mulig.

Ikke skyt spurv med kanon

Små lokale modeller til avgrensede oppgaver,
store modeller der de trengs.

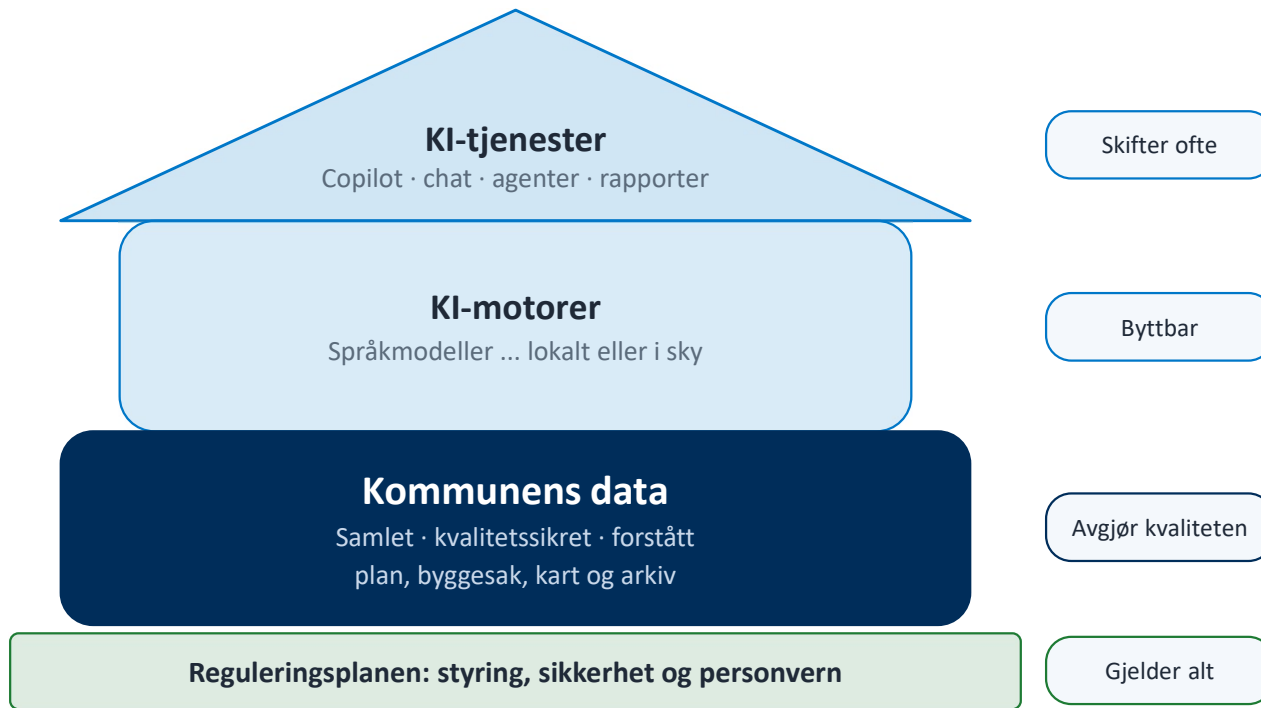
Valgt per bruksområde.

Tokenpris er ikke TCO

Drift, kompetanse, sikkerhet, utvikling,
observability, støtte og skalering kommer i tillegg.

KI er ikke en knapp | Det er et bygg.

Kommunen må **eie** grunnmuren selv | **før** vi bygger fasaden.



«Modellene kommer til å skifte.

Kommunens data og betydningen av dem er den langsiktige verdien.»

Vi eier grunnmuren, leier resten

- Dårlige data blir bare raskere dårlige ... KI forsterker kvaliteten, på godt og vondt
- Modeller og leverandører skifter. Dataene består.

Vi bestemmer hva som forlater huset

- Sensitive data blir hos oss. Sky der det er trygt og lønner seg.

Smått først, så skalere

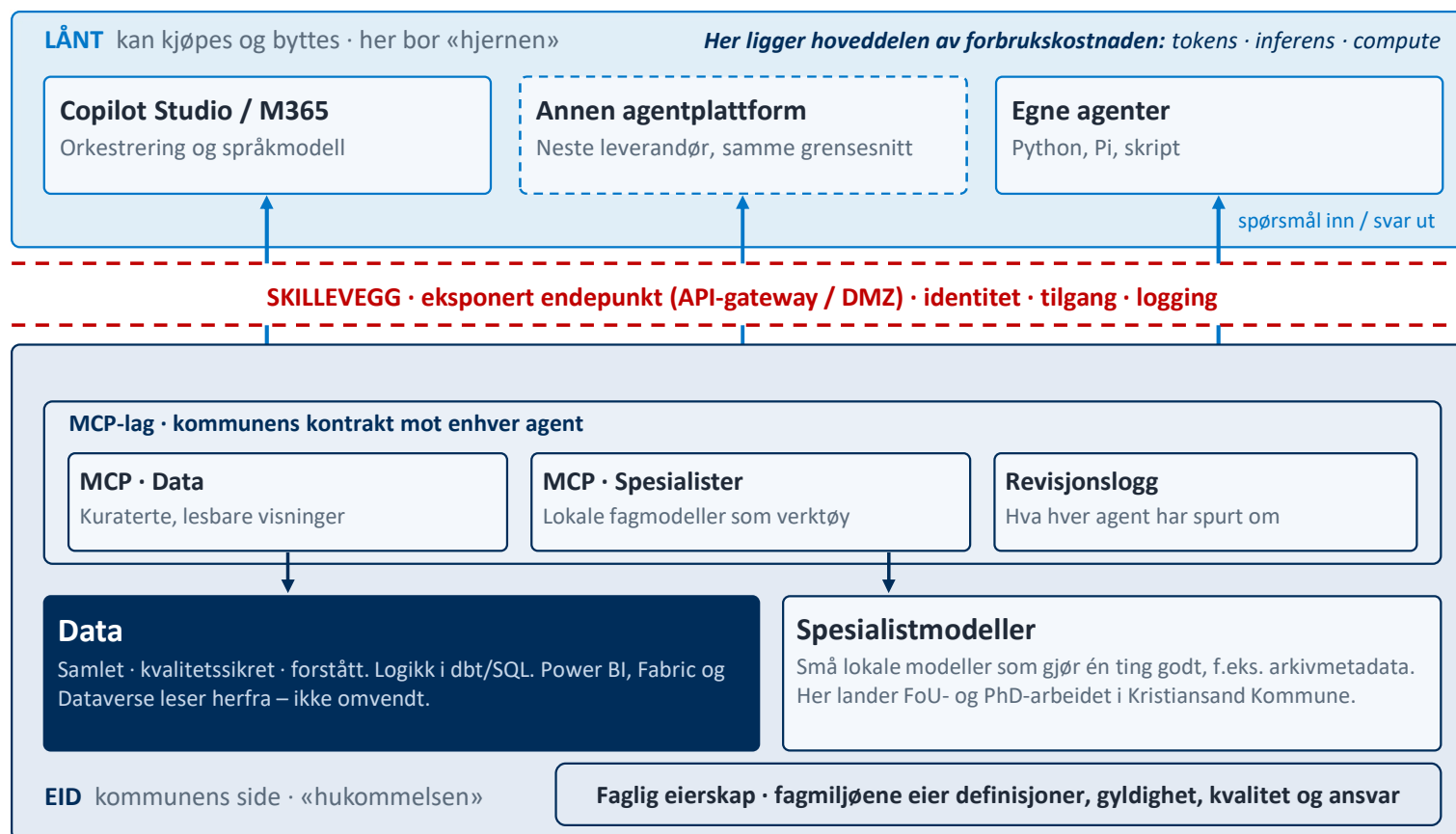
- Demonstrert i lab før større investeringer.

HVA DET BETYR

- ✓ Raskere og bedre beslutningsgrunnlag i saksbehandlingen
- ✓ Plan- og kartdata som KI faktisk forstår
- ✓ Mindre manuelt søk og sammenstilling
- ✓ Svar til innbyggere som henger sammen

Lån hjernen, eie hukommelsen

Inferens og orkestrering kan kjøpes og byttes. Data, logikk og spesialistverktøy forblir kommunens.



TEGNFORKLARING

- Lånt – kan kjøpes og byttes
- Eid – kommunens kontroll
- Kontrollpunkt (designkrav)

Tenkt modell: all trafikk til underliggende lag går via API/MCP med nøkler. Brukertilgang styres i agentplattformens eget økosystem.



«Leverandørene får låne oss hjernen ... men aldri eie hukommelsen.»

Bruk markedet | uten å overlate arkitekturen til markedet

Pragmatisk, ikke anti-sky eller anti-leverandør.

Kommunen kontrollerer

- Data og betydningen av dem
- Logikk i dbt/SQL ...
ikke i leverandørens verktøy
- Integrasjoner og MCP-grensesnitt
- Agentoppsett og konfigurasjon i Git
- Exit-krav i avtalene

Kan deles i sektoren

- Modelltilgang
- Gateway-standard og delte komponenter
- Konfigurasjon som kode, uten persondata
- Referansemiljø som sandkasse

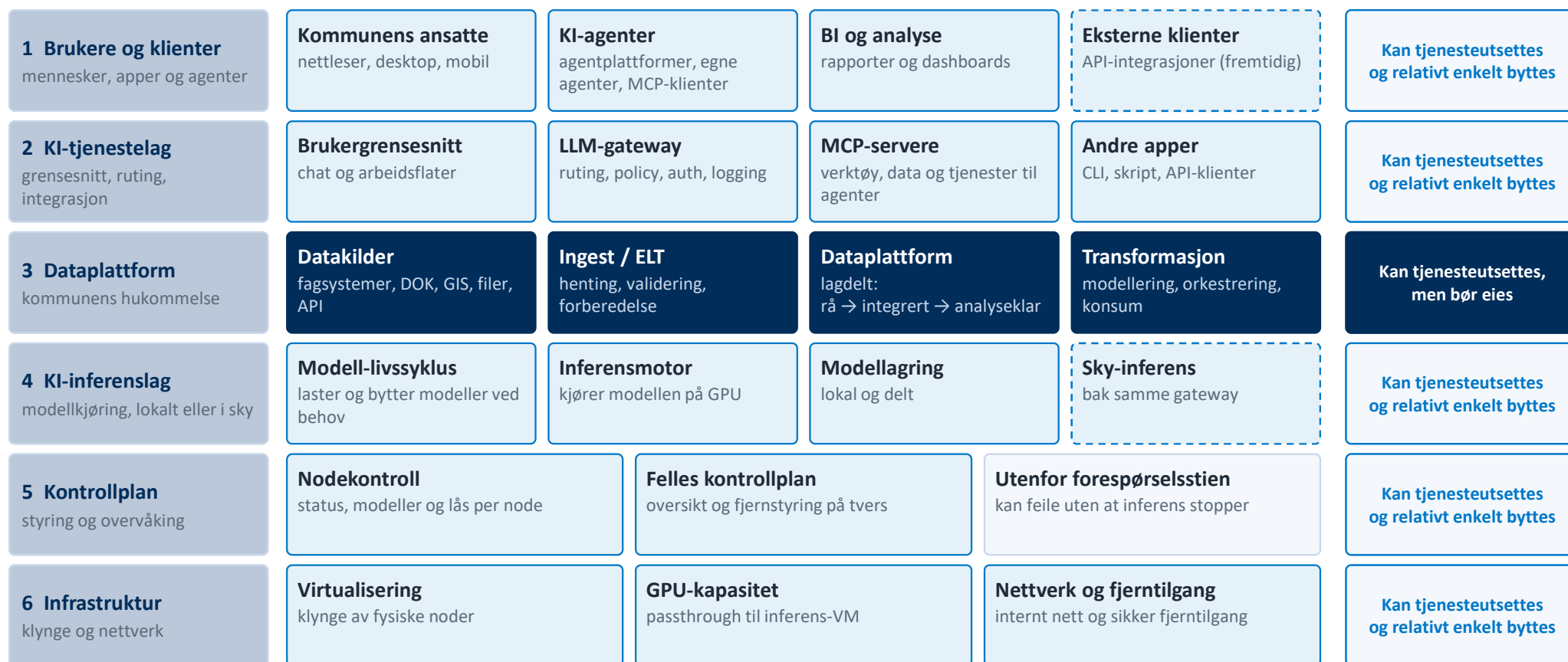
Markedet leverer

- Modeller og sky-inferens
- Assistenter og agentflater, f.eks. Copilot
- Hosting og plattformkomponenter
- Drift og FDV ved skalering

Strategisk autonomi er ikke selvbygging som mål. Det er kontroll over kritiske data, kontrakter, arkitektur og avhengigheter.

Seks ansvarsområder, løst koblet

Kategorier og ansvar | uavhengig av produktvalg.



Mørk blå = bør eies av kommunen · lys blå = kan tjenesteutsettes og byttes · stiptet = valgfritt / fremtidig

Samme arkitektur | med dagens valg

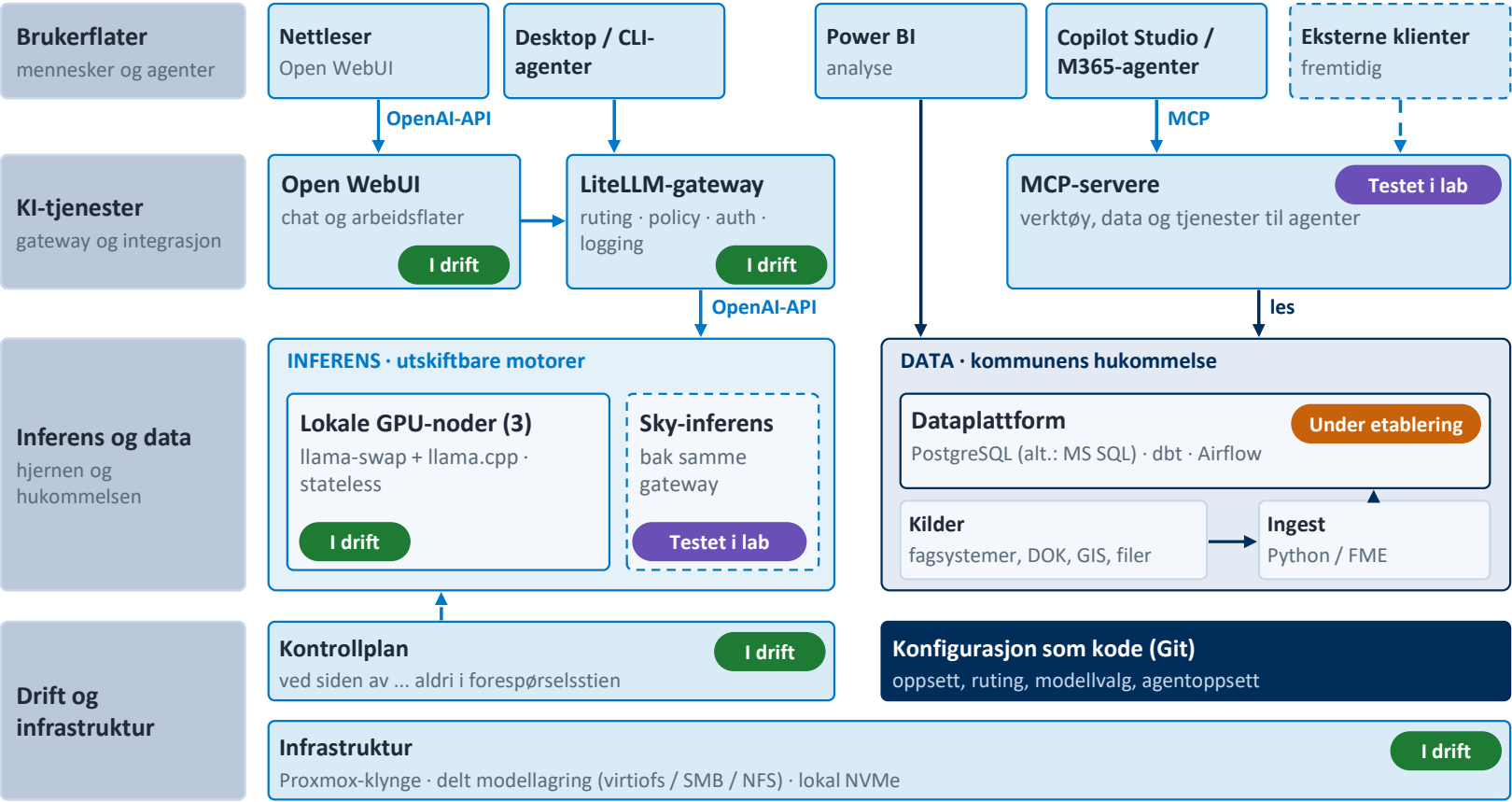
Produktene er valg vi har tatt nå. Nesten alle blokker har alternativer og kan byttes uten å bygge om resten.

1 Brukere og klienter mennesker, apper og agenter	Nettleser · mobil via Open WebUI	Codex · Claude · Pi agentrammeverk mot egen inferens	Copilot Studio M365-agenter	Power BI rapporter mot analyseklare data	Kan tjenesteutsettes og relativt enkelt byttes
2 KI-tjenestelag grensesnitt, ruting, integrasjon	Open WebUI alt.: annet OpenAI-kompatibelt UI	LiteLLM nøkler og budsjetter mulig alt.: Bifrost	MCP-servere Python / FastAPI · enkle verktøy	CLI-verktøy, skript API-klienter m.m.	Kan tjenesteutsettes og relativt enkelt byttes
3 Dataplattform kommunens hukommelse	Fagsystemer DOK, GIS, filer, IoT, eksterne API-er	Python · FME skript og integrasjoner	PostgreSQL bronse → sølv → gull alt.: MS SQL	dbt · Airflow Power BI og MCP konsumerer	Kan tjenesteutsettes, men bør eies
4 KI-inferenslag modellkjøring, lokalt eller i sky	llama-swap på hver node (3 noder)	llama.cpp · Strata alt.: Ollama, vLLM, SGLang	NVMe + delt lager virtiofs / SMB / NFS	Sky-enderpunkter OpenAI-kompatible, ikke i aktiv bruk pt.	Kan tjenesteutsettes og relativt enkelt byttes
5 Kontrollplan styring og overvåking	Per node web-UI (PIN) + REST-API · lås	Master (kpve0) metrikk, modellmatrise, hendelseslogg	Ikke i stien kan feile eller byttes uten at inferens stopper		Kan tjenesteutsettes og relativt enkelt byttes
6 Infrastruktur klynge og nettverk	Proxmox-klynge kpve0 · kpve1 · kpve2 alt.: annen virtualisering	GPU-noder passthrough til inferens-VM	Lab-LAN · Tailscale fjerntilgang		Kan tjenesteutsettes og relativt enkelt byttes

Dagens valg · *alt.: alternativ* | ● i drift ● testet ● under etablering ● neste | stiple = valgfritt / ikke i aktiv bruk

Slik henger byggeklossene sammen

Hvem snakker med hvem | og gjennom hvilket grensesnitt.



E I E R S K A P

Kan tjenesteutsettes og relativt enkelt byttes

Kan tjenesteutsettes, men bør eies

Valgfritt / fremtidig

S T A T U S

I drift

Testet i lab

Under etablering

Neste

«Proper loose coupling ...
not merely
«open source good».»

Åtte regler som holder døren åpen

Uansett hvilken leverandør som velges. **Kravet stilles til grensesnittet**, ikke produktet.

1

Én sannhet i dataplattformen

Power BI, Fabric og Dataverse leser fra den ... aldri omvendt

2

Spesialistmodeller som verktøy

Lokale fagmodeller eksponeres via MCP til enhver agentplattform

3

Transformasjoner i dbt/SQL

Ikke i Data Factory eller Dataflows ... plattformen under kan byttes

4

Agentoppsett i Git

Instruksjoner, prompter og oppsett versjoneres
... gjenskapbart, ikke flyttbart

5

Tykke data, tynne agenter

Kunnskap og logikk ligger i data- og MCP-laget

6

Exit er et krav

Avtalen må vise hvordan data, integrasjoner og agentlogikk tas med ut

7

MCP er kontrakten

Kuraterte visninger, tilgang per agent eller bruker, kommunen eier loggen

8

Bevisst om hva som forlater huset

Svar havner i leverandørens kontekst ... avgjøres per datakategori

Grensesnitt reduserer koblinger, men løser ikke alt:

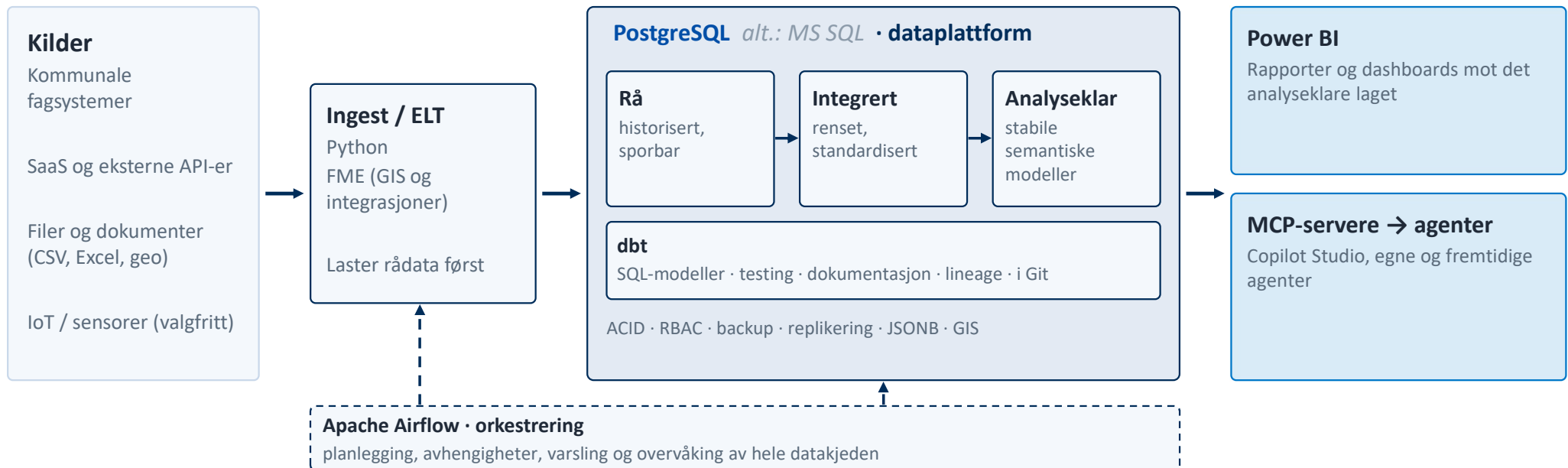
MCP løser ikke autorisasjon av seg selv

- OpenAI-kompatibelt API gir ikke full funksjonell eller semantisk portabilitet mellom modeller.

Dataene er sentrum | ikke KI-modellen

Under etablering

Foreløpig målarkitektur · etablerte mønstre fra datavarehus og analyse ... ikke noe nytt

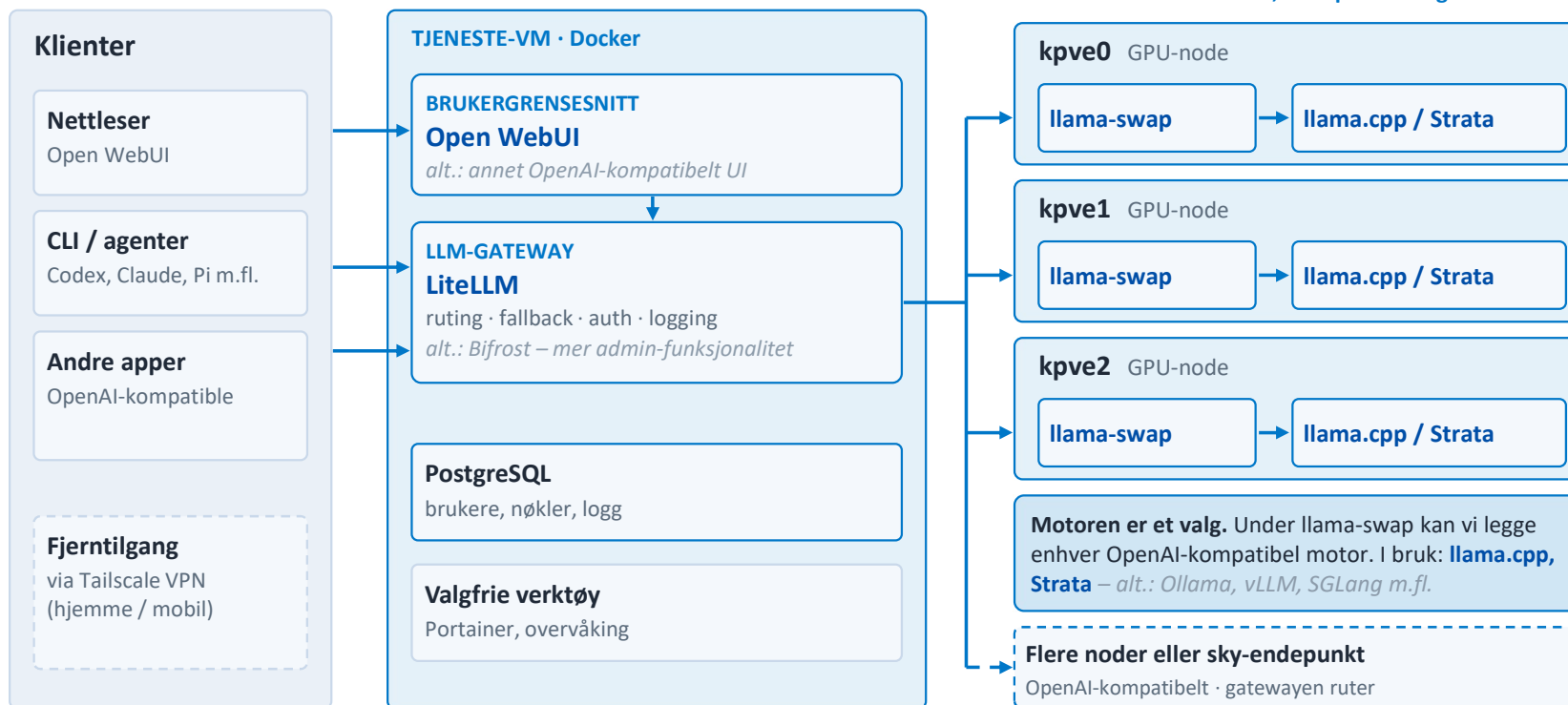


KI gjør ikke dårlige data gode.

Den gjør det bare mulig å bruke dårlige data raskere ... og i større skala.

Inferensen kjører | og hver blokk er et valg

I drift

Referanseimplementering · tre lokale GPU-noder bak én gateway · **dagens valg i blått** · *alternativer i grått***Svarer på:***Hvor skal forespørselen?***Svarer på:***Hvordan kjører jeg modeller effektivt på denne maskinen?*

KONTROLLPLAN ved siden av stakken ... bevisst ikke i forespørselsstien.
Kan feile eller byttes uten at inferens stopper.

Valg, ikke låsing

Rollene er arkitekturen. Produktene er dagens valg ... og kan byttes uten å røkke båten vesentlig.

Stateless

Inferens-VM-ene lagrer ikke prompts, svar eller samtaler.

Historikk og logging håndteres i tjenestelaget.

«Run the right model, on the right node, at the right time.»

Hva som faktisk er realisert

Skillet mellom bygget, under arbeid og ikke bevist skal være tydelig.

I drift

Referansen, 1–5 brukere

- Inferens på tre lokale GPU-noder
- Gateway og brukergrensesnitt
- Kontrollplan
- Benchmarking
- Agentrammeverk mot egen inferens

Testet i lab

Testet og verifisert

- Sky-endepunkter bak gatewayen
- MCP med enkle verktøy
- Kø og parallellitet på nodene
- API-nøkler og brukerkontoer

Under etablering

Bygges nå

- Dataplattform: PostgreSQL, dbt, Airflow
- Microsoft-sporet: Copilot Studio, Dataverse, Fabric
- Kartlegging av mulighetsrom

Neste

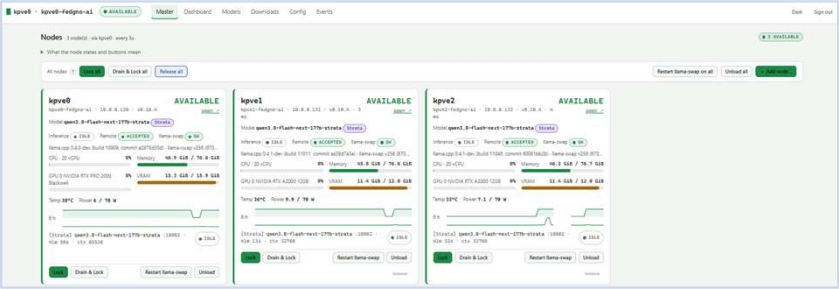
Står for tur

- MCP mot reelle data ... når første pipelines er på plass
- Agentlag (Copilot)
- Første fagagent: byggesak

Ikke verifisert

Ikke testet ennå

- Kapasitet ut over 1–5 brukere
- SSO og tilgangskontroll i hele kjeden
- Drift, FDV og SLA i kommunal skala



Kontrollplanet i drift
Status for noder, GPU og modeller.
Skjerm bilde fra referanseimplementeringen.
Flere skjermbilder og benchmarks ligger i appendiks.

Små leveranser, korte læringssløyfer

Forankret internt og løpende justert. Innholdet i hvert steg detaljeres når vi kommer dit.

Arbeidsplan



Hvert steg: bygg, ta i bruk, lær – og juster neste steg.

Enkelt?

Enkel å designe?

Nei.

Enkel å forstå?

I økende grad, ja.

Enkel å reprodusere?

Potensielt i stor grad, ja.

Enkel å drifte uten kompetente folk?

Absolutt ikke.

«Dette er resultatet av en iterativ FoU-prosess der vi har bygget akkurat nok til å teste neste problemstilling. Etter hvert har det vokst frem en sammenhengende arkitektur.»

«Vi har ikke skalert kostnaden først. Vi har skalert arkitekturen først.»

«Vi har redusert teknisk kompleksitet – ikke behovet for kompetanse.»

Fra referanse til skalert drift

Hva som er demonstrert, hva som forventes å skalere, og hva som må løses.

Demonstrert i referansen

- Byttbare modeller bak et standard grensesnitt
- Inferens skilt fra data og applikasjoner
- Drift og kontroll ved siden av, aldri i forespørselsstien
- Noder som brukes alene eller samles i en felles pool
- Sky-endepunkter bak samme gateway (testet)

Skalerer med kapasitet

- Flere og større GPU-noder
- Motorbyte ved behov (f.eks. vLLM, SGLang)
- Kø og parallellitet per node
- Lagring og nettverk

Forventet ut fra designet – ikke testet ved skala

Må løses før bred produksjon

- Identitet og tilgang (SSO) gjennom hele kjeden
- Skillevegger for data og tilgang
- Logging, sporbarhet og KI-forordningen
- Forvaltning, drift og vedlikehold (FDV)

Det som gjenstår kan godt leveres av markedet. Målet er at arkitekturen gjør det utskiftbart – ikke vanskeligere.

En målestokk for markedsdialog

Koblet til temaene i KS' markedsdialog om felles KI-satsing.

Tema	KS spør	Svar fra arkitekturen
Omfang	Hva løses nasjonalt, av markedet og lokalt?	Nasjonalt: modelltilgang, gateway-standard, delte komponenter. Markedet: modeller, sky-inferens, assistenter. Lokalt: dataplattform og egne noder.
Arkitektur	Hvordan organisere data for deling?	Data først: lagdelt dataplattform med tydelig semantikk og eierskap.
Arkitektur	Teknologinøytrale krav til API-er?	OpenAI-kompatibelt API mot modeller, MCP mot verktøy og data. Kravet stilles til grensesnittet, ikke produktet.
Arkitektur	Bytte leverandør uten å miste integrasjoner?	Integrasjonene peker mot gatewayen, ikke leverandøren. Stateless inferens, data i egen plattform.
Modeller	Flere modeller og bytte over tid?	Demonstrert: flere modeller per node og ruting mellom noder. Sky-endepunkter bak samme gateway er testet.
Modeller	Suverenitet og kontroll over egne data?	Sensitive data kan bli lokalt; sky der det er trygt og lønner seg. Valget tas per bruksområde.
Data og ansvar	Hva skjer når avtalen avsluttes?	Inferensnodene lagrer ingen samtaler. Dataene ligger i kommunens egen plattform.
Deling	Sandkasse og deling uten persondata?	Referansemiljøet kan brukes som sandkasse. Oppsett deles som kode, og resultater publiseres åpent på GitHub.

Tre ting å ta med

1

Lån hjernen, eie hukommelsen er et arkitekturprinsipp: modeller og plattformer byttes – data og betydningen av dem består.

2

Kravet stilles til grensesnittet, ikke produktet. Og exit er et krav i enhver avtale.

3

Bygg akkurat nok, lær raskt – og vær ærlig om hva som ikke er bevist ennå.



Rød eller grønn pille?

Diskusjon: Hva bør løses nasjonalt, hva av markedet – og hva må hver kommune eie selv?



Kristiansand
kommune

Rune Ødegård

Rådgiver | Plan og bygg | Kristiansand kommune

Tlf. 97 97 50 77

Rune.Odegard@Kristiansand.Kommune.no

labs.geokrs.no

linkedin.com/in/runeodegard

Kan i prinsippet spres fritt med tydelig kildehenvisning til Kristiansand kommune. Deles ikke utenfor offentlig/kommunal sektor uten forespørsel til Kristiansand kommune.

NERDING

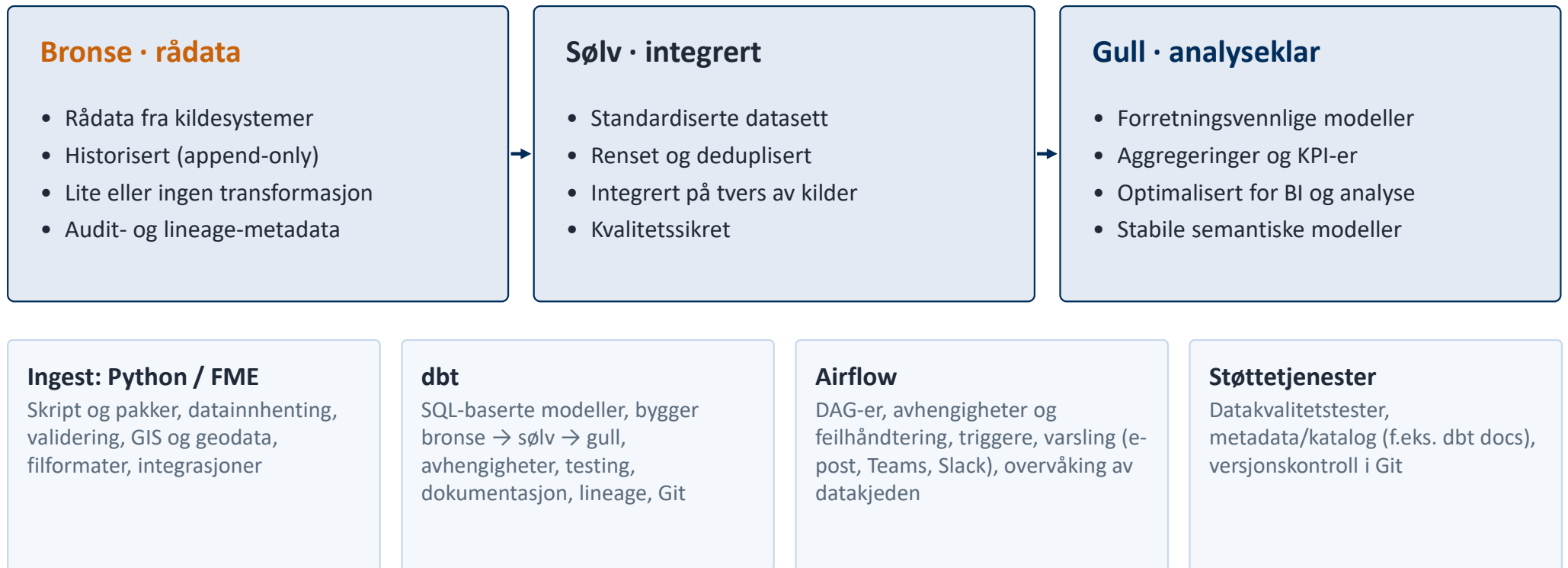
Appendiks

For faglig etterprøving – data, implementering, benchmark og labmiljø

Lagdeling i dataplattformen (medallion)

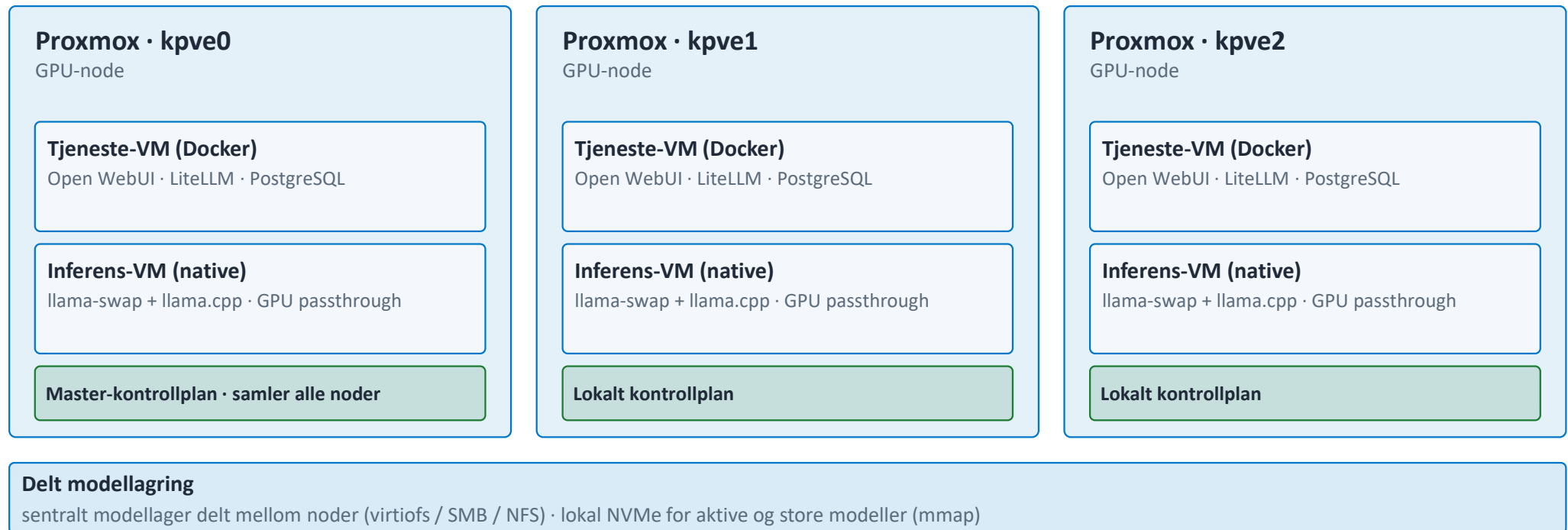
Bronse → sølv → gull, transformert i dbt og orkestrert i Airflow.

Under etablering



Fysisk implementering i referansen

Tre Proxmox-noder. Hver node er en selvstendig celle – samlet i en felles pool.



Celle ↔ pool: hver node kan brukes alene. LiteLLM på kpve0 samler alle. Lås via kontrollplanet (UI/API) gir eksklusiv bruk.

Kontrollplanet i drift

Skjermbilder fra referanseimplementeringen.

I drift

▼ Model matrix

every model on every node - copy between nodes

Filter models...

☐ only models missing somewhere or differing

11 model(s) on 1 node(s)

Refresh

MODEL	KPVE0 15.9 GiB VRAM	KPVE1 12.0 GiB VRAM	KPVE2 12.0 GiB VRAM
Qwen3.8 27B AP IQ4_XS ap-qwen3.8-27b-ap-1q4-xs.gguf / qwen3.8-27b-ap-1q4-xs.gguf	2025-09-23 10.3 GiB [RECEIVED] used 2025-10-02 [Link: copy] 48.7 t/s 110ms 120s	- copy here...	- copy here...
Qwen3.8 27B Heretic NEO CODER IQ3_M her-qwen3.8-27b-heretic-neo-coder-1q3-m.gguf / qwen3.8-27b-heretic-neo-coder-1q3-m.gguf	2025-09-14 11.1 GiB [RECEIVED] used 2025-10-02 [Link: copy]	2025-09-17 11.1 GiB [RECEIVED] used 2025-10-28 [Link: copy] 6.3 t/s 110ms 60s	- copy here...
Gemma 3 4B Q4_K_M gemma-3-4b-q4-k-m.gguf / [Downloaded on different device (same size)] gemma-3-4b-q4-k-m.gguf	2025-09-25 1.1 GiB [RECEIVED] used 2025-10-02 [Link: copy]	2025-09-29 1.1 GiB [RECEIVED] used 2025-10-28 [Link: copy] 66.6 t/s 110ms 60s	- copy here...
Huahai Qwen3.8 27B Abliterated IQ3.5 huahai-qwen3.8-27b-abliterated-1q3.5.gguf / huahai-qwen3.8-27b-abliterated-1q3.5.gguf	2025-09-10 22.6 GiB [RECEIVED] used 2025-10-02 [Link: copy]	- copy here...	- copy here...
Qwen3.8 Flash-Next 177B IQ3_XOS - Strata 1q3_xos / qwen3.8-flash-next-177b-iq3-xos-strata-1q3_xos.lcp113	2025-10-03 70.6 GiB [RECEIVED] used 2025-10-02 [Link: copy] 48.8 t/s stream 60s	2025-10-03 70.6 GiB [RECEIVED] used 2025-10-28 [Link: copy] 46.2 t/s stream 60s	2025-10-03 70.6 GiB [RECEIVED] used 2025-10-28 [Link: copy] 48.1 t/s stream 60s
Qwen3.8 9B OpusGLM Abliterated Q8 op-qwen3.8-9b-opusglm-abliterated-1q8.gguf / q3.8-9b-opusglm-max-9b-iq3-abliterated-1q8.gguf	2025-09-14 9.9 GiB [RECEIVED] used 2025-10-28 [Link: copy]	2025-09-14 9.9 GiB [RECEIVED] used 2025-10-28 [Link: copy] 43.5 t/s 110ms 120s	- copy here...
qwen-image-2.1-uc-q4_k_m qwen-image-2.1-uc-q4-k-m.gguf	2025-09-25 4.3 GiB no entry	- copy here...	- copy here...
Qwen3.8 27B G5Q RCO IQ3.5 MTP qwen3.8-27b-g5q-rcq-iq3.5-mtp.gguf / qwen3.8-27b-g5q-rcq-iq3.5-mtp.gguf	2025-09-18 11.3 GiB [RECEIVED] used 2025-10-28 [Link: copy] 27.8 t/s 110ms 60s	- copy here...	2025-09-18 11.3 GiB [RECEIVED] used 2025-10-28 [Link: copy] 6.8 t/s 110ms 120s
Qwen3.8 Flash-Next 177B IQ2_K_XL qwen3.8-flash-next-177b-iq2-k-xl.gguf / qwen3.8-flash-next-177b-iq2-k-xl-g1e1a1a1.gguf	2025-09-16 71.3 GiB [RECEIVED] used 2025-10-01 [Link: copy]	- copy here...	2025-09-16 71.3 GiB [RECEIVED] used 2025-10-27 [Link: copy] 3.8 t/s 110ms 60s
Qwen3.8 9B Q6_K_XL qwen3.8-9b-q6-k-xl.gguf / qwen3.8-9b-q6-k-xl.gguf	- copy here...	2025-09-14 12.2 GiB [RECEIVED] used 2025-10-28 [Link: copy] 48.7 t/s 110ms 120s	- copy here...
Qwen3.8 27B IQ3_XOS qwen3.8-27b-iq3-xos.gguf / qwen3.8-27b-iq3-xos.gguf	2025-09-14 11.0 GiB [RECEIVED] used 2025-10-02 [Link: copy]	- copy here...	- copy here...



Venstre: modellmatrise på tvers av noder, hvilke modeller som er registrert, lastet og passer i VRAM.

Høyre: CLI-visning med sanntids GPU-, VRAM- og inferensstatus per node.

Benchmark 7. oktober 2026

Tall fra spesialskevne, standardiserte tester per node. Tolkning AI-assistert. Lab-indikasjon.

	kpve0	kpve1	kpve2
Decode (tokens/s)	61	38	39
Prefill, 8k prompt (tokens/s)	ca. 1 230	ca. 790	ca. 750
Kontekstlengde	65 536	32 768	32 768
Krasj under ca. 45 min last	0	0	3

Én 177B MoE-modell (kvantisert) fordelt på tre noder.

Bruk: FoU, modelleksperimenter, agentorkestrering og batch – ikke tjeneste for mange samtidige brukere.

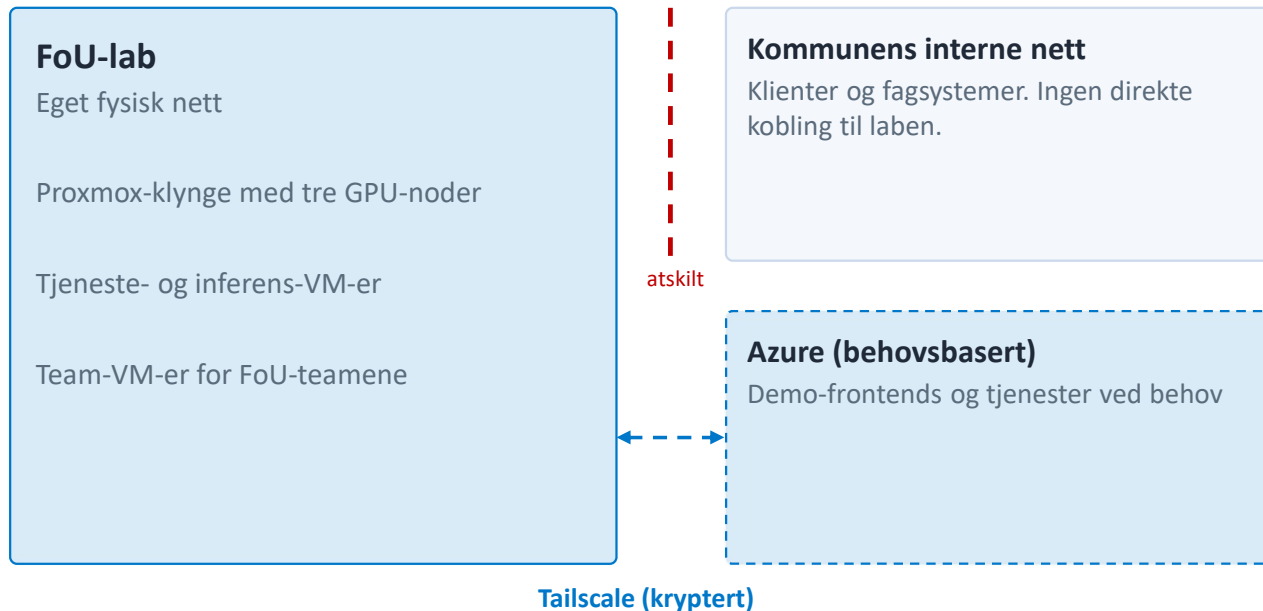
Hva det viser

- Føderasjon: ca. 96–104 tokens/s mot ca. 120 teoretisk tak (80–86 %)
- Svartid per strøm i samme klasse som sky-API-er

Hvor det er lite

- Samtidighet 1 per node i målingen – kø og parallellitet ikke aktivert
- Én modell, tre kopier – ingen modellbredde
- Stabilitet ikke på produksjonsnivå (kpve2)
- Kvalitet: god, men ikke frontier-nivå

FoU-lab, atskilt fra kommunens nett

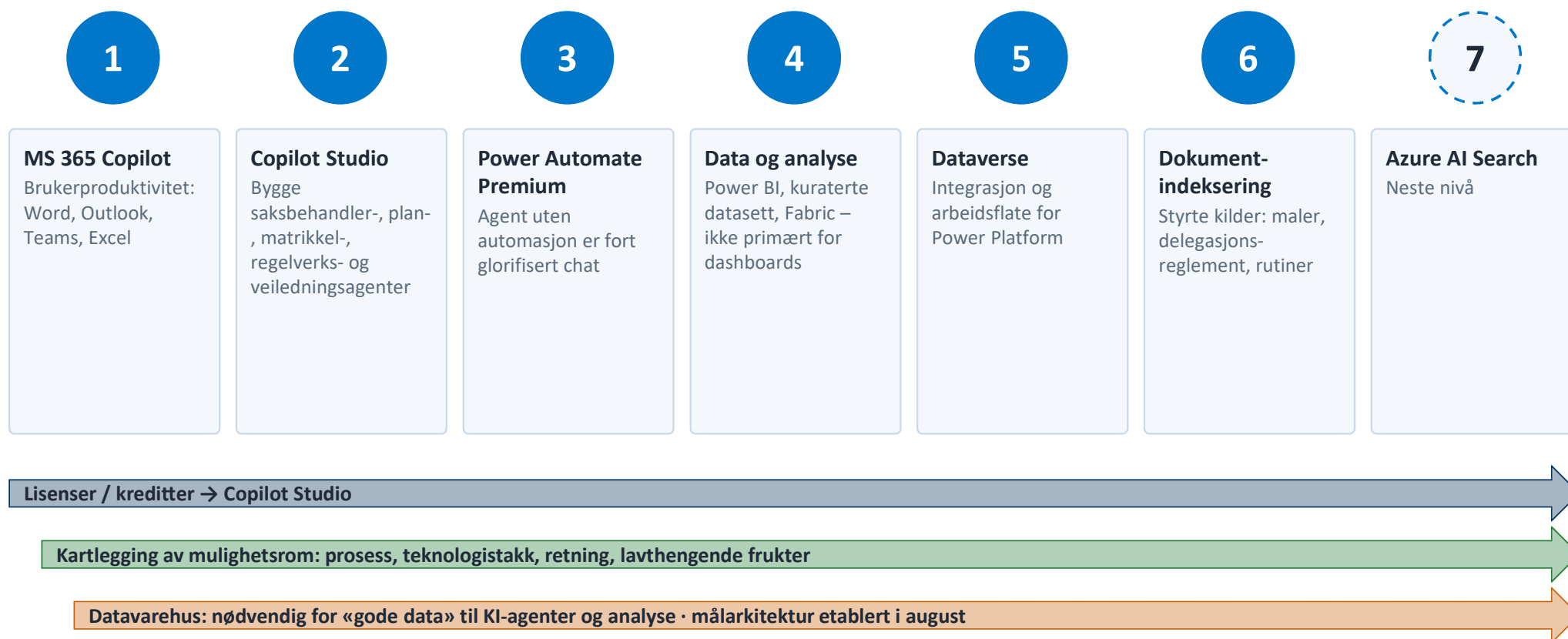


- Fysisk og logisk atskilt fra kommunens nett
- Fjerntilgang via Tailscale: CORE for drift, PUBLIC for tjenester til FoU-teamene
- Azure brukes behovsbasert for demo-frontends
- Samme mønster lokalt og i sky – komponentene kan flyttes

Lab ≠ produksjon. Produksjon kan kjøres lokalt, i sky eller hybrid – ikke utredet ennå. Arkitekturen overføres gradvis til hybrid drift, og noder kan flyttes inn i produksjonsnettets sammen med IT.

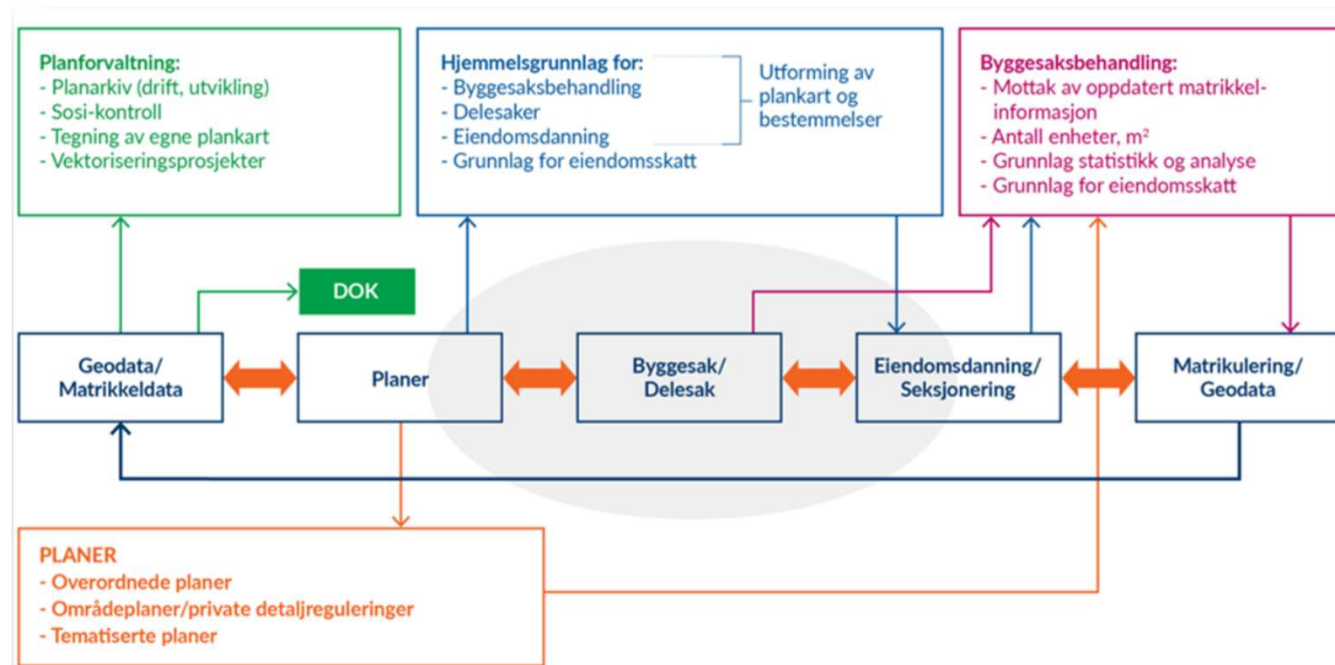
Modningsnivåer for KI-agenter i M365-økosystemet

Parallelt spor i Plan og bygg – kobles til arkitekturen via MCP og dataplattformen.



Hvor agentene skal virke i Plan og bygg

Verdikjeden fra geodata og planer til byggesak, eiendomsdanning og matrikkel.



Agent 1 · Byggesaksveileder

Spørsmål, rutiner, maler → forslag og veiledning. Kun RAG, ingen integrasjoner.

Agent 2 · Eiendomsdanning

Behov detaljeres når vi kommer dit (smidig)

Agent 3 · Matrikkel

Matrikkel, rutiner, håndbøker

Agent 4 · Plan

Planer, bestemmelser, planarkiv – forklare planstatus og bestemmelser

Kartlegging av mulighetsrom kommer før agent 1: prosess, teknologistakk, retning og lavhengende frukter.