

ERFARINGER MED Å BYGGE EN MODULÆR MONOLITT

Torstein Nicolaysen ▪ Ensō og NNUG-styret ▪ 29. oktober 2024

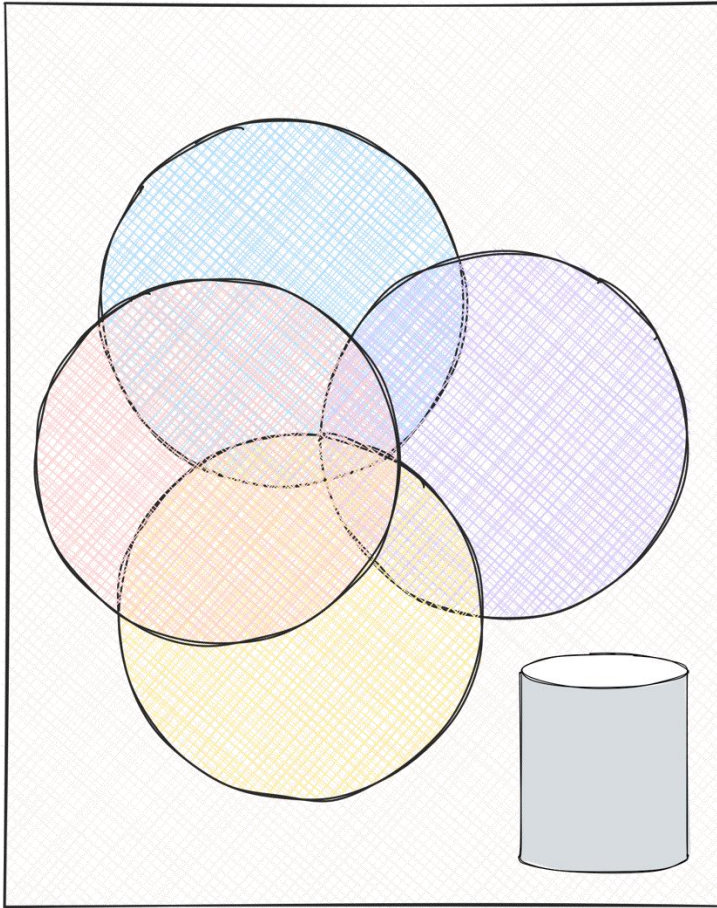


ensō

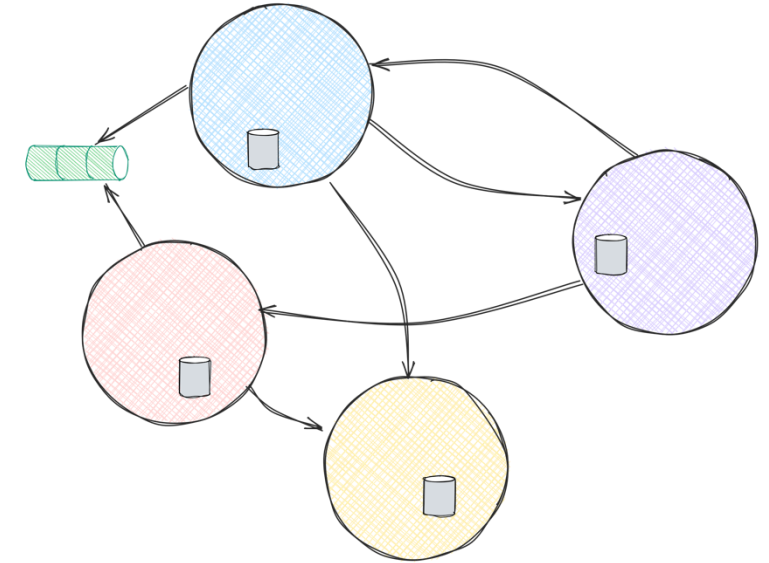
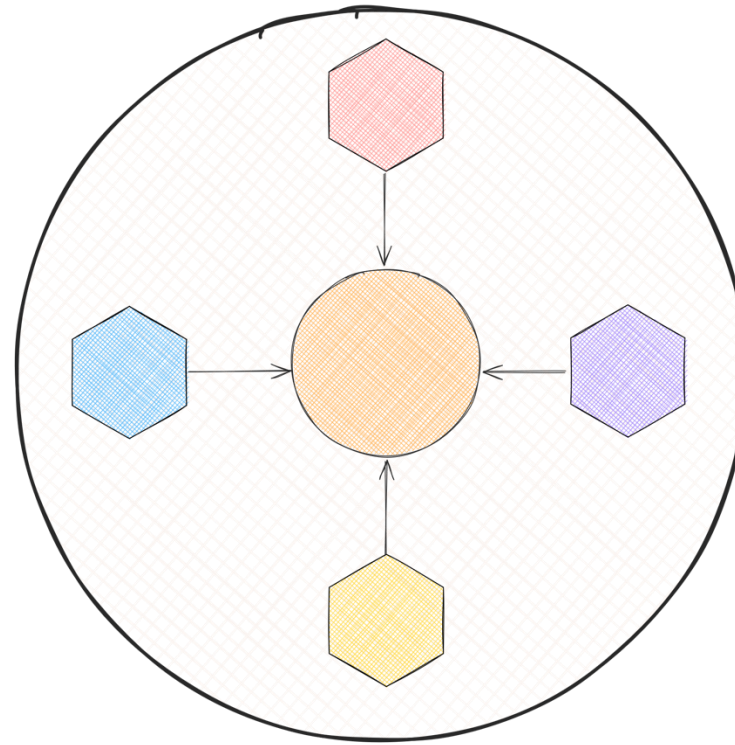
KORT INTRO TIL MODULÆR MONOLITT

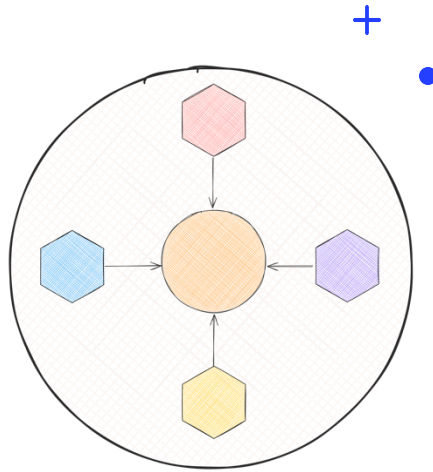


MONOLITT



MIKROTJENESTER





MODULÆR MONOLITT

Beste av begge verdener(?)

Pragmatisk

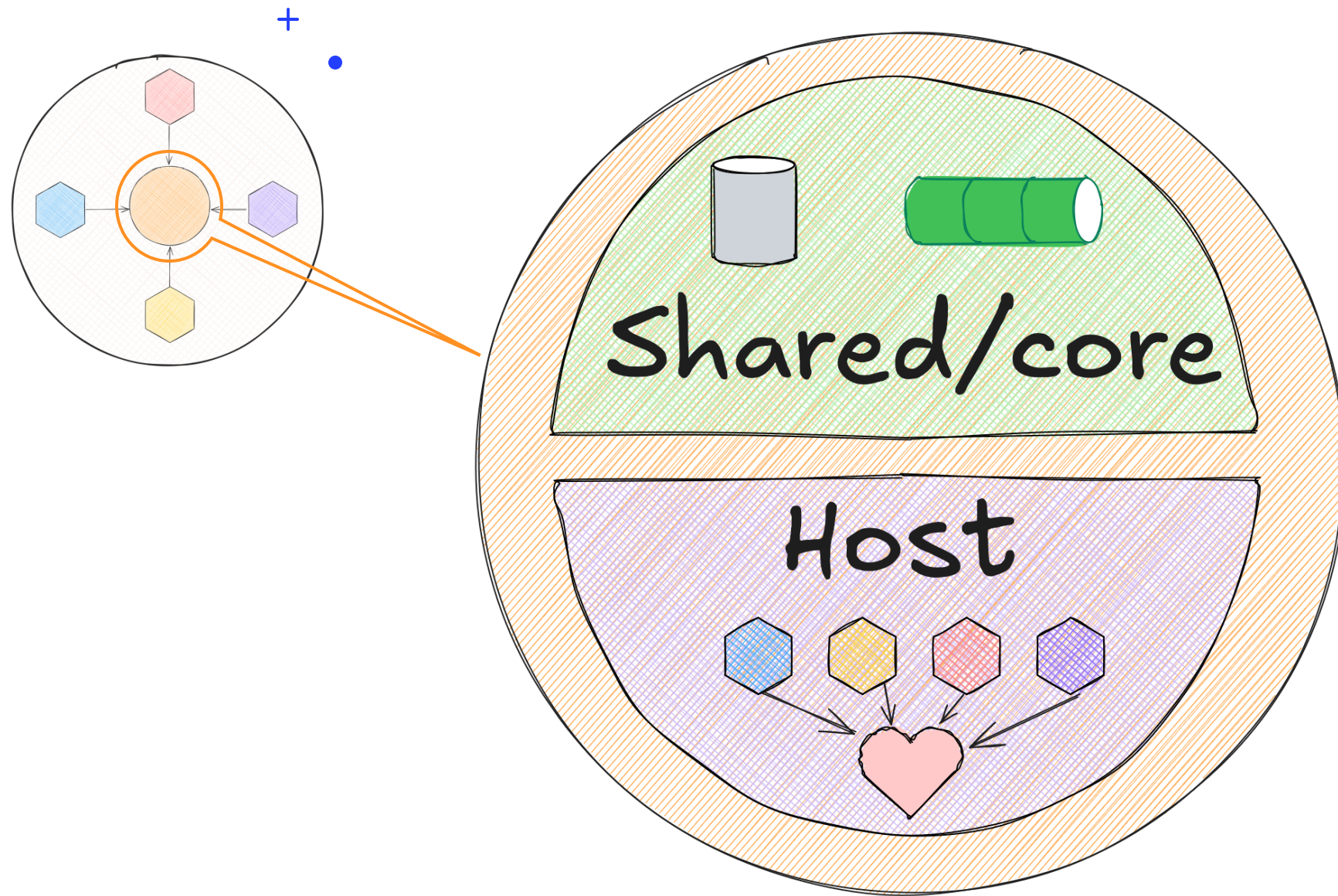
Valgfrihet på løsninger

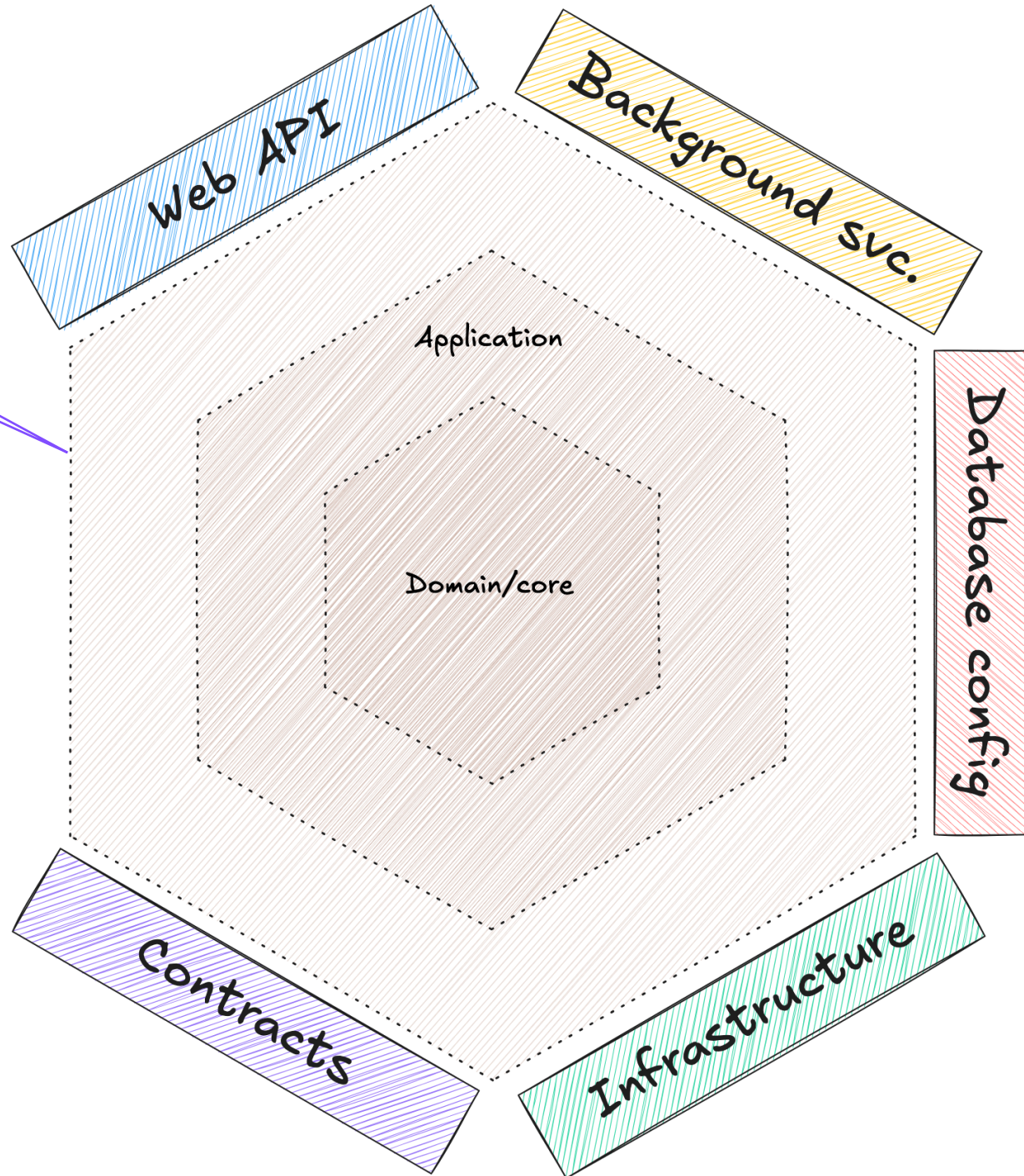
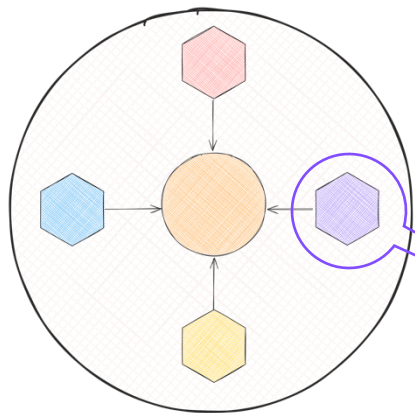
Alt kjører i samme prosess

Vedlikeholdbart

Enkel utrulling

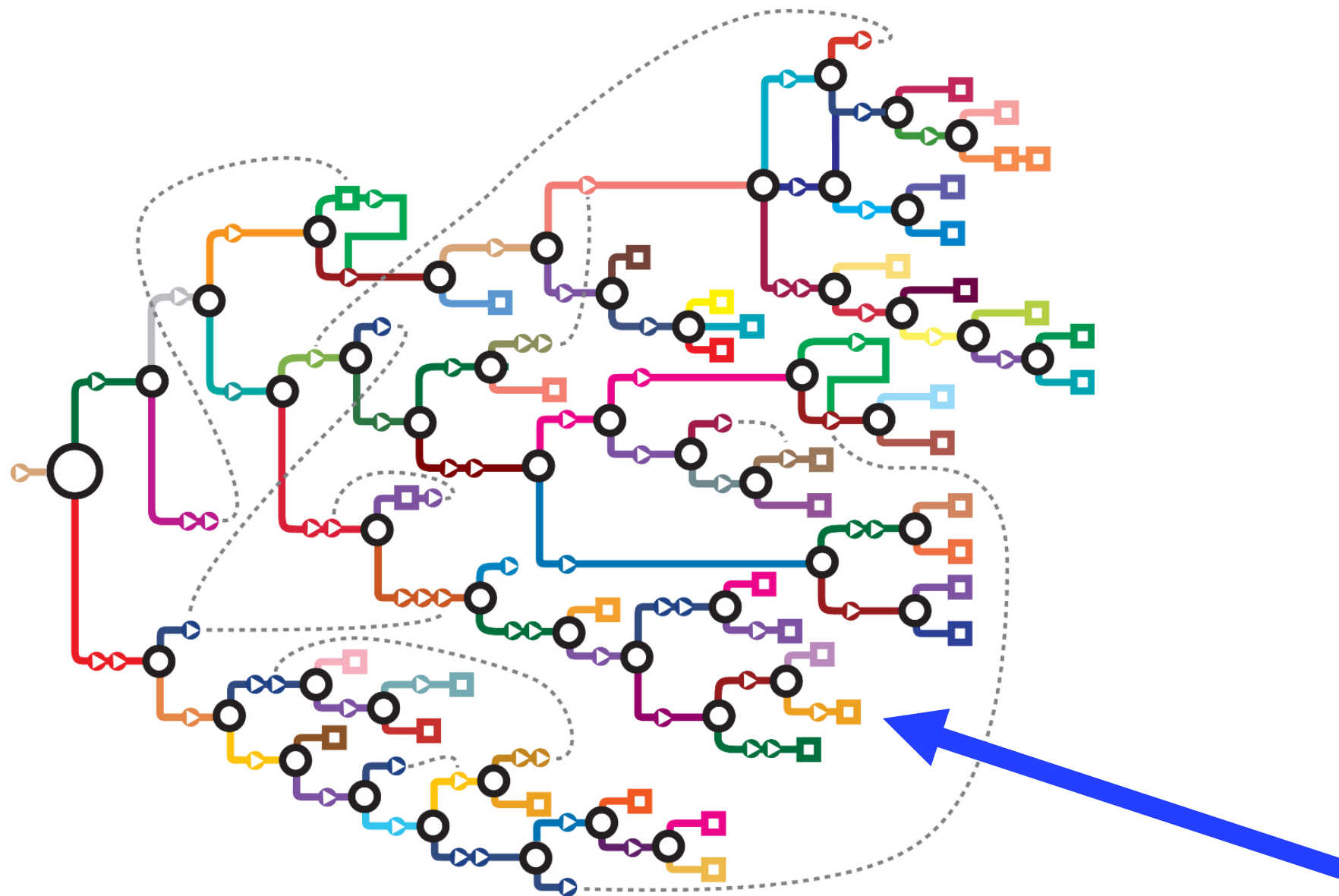
Skalerbart





TIPS FOR HVORDAN BYGGE EN MODULÆR MONOLITT



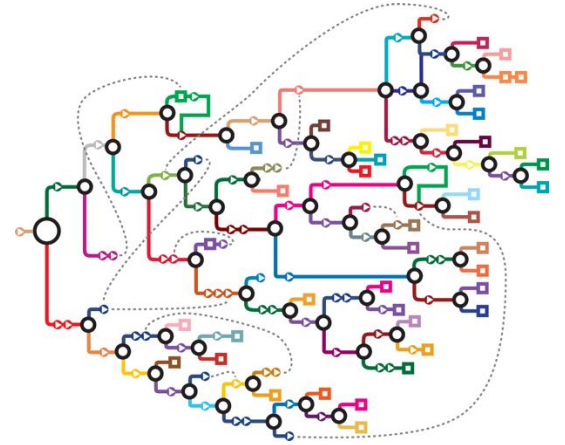


TEAMET

- Mod. monolitt eies av kun et team
 - Et team kan eie flere (ved behov)
- En person/par-prog. team per modul på et gitt tidspunkt
- Planlegg og koordiner endringer i kjernen eller på tvers av moduler
- Kontinuerlig kompetansedeling
 - Tech talks
 - Par-programmering
 - Pull-requests
 - Aktiv deling av kunnskap på chat

ARKITEKTUR OG DESIGN

- «Architectural decision record» (ADR)
- Arkitekturskisser (gjerne med C4-modellen)
- Viktig å finne riktig «grenser» for modulen
- Ikke vær redd for å revurdere og redesigne underveis
- Finn ut om det er eksterne konsumenter
- Identifiser samhandling med andre moduler tidlig
- En modul bør kunne enkelt «klippes ut» og kjøre selvstendig



TEKNISKE VALG

- Arkitekturtester ([ArchUnitNET](#)) og verifikasjonstester
- God kodevalitet er viktig
 - .editorconfig (med code style og analyzer-oppsett)
- Ikke bruk interne domene-objekter i grensesnitt
 - Bruk dedikerte kontrakts-objekter
- Modulene bør kjenne minst mulig til detaljer om teknologien rundt

DATABASEDESIGN

- Samme prinsipper som koden – en database, mange schemas
 - Schema per modul
 - Tenk på det som et «namespace» eller «inngjerding» av tabeller som eies av en modul
- Ikke lag spørringer som går på tvers av schemas
 - Finnes løsninger hvis det må gjøres (f.eks. pga. ytelse eller kompleksitet)
- Det bør være mulig å flytte en moduls tabeller til en ny database
- Kan ved behov ha forskjellige lagrings-teknologier per modul

HVORFOR JEG ANBEFALTE EN MODULÆR MONOLITT TIL KUNDEN

Litt om hvordan du kan vurdere om en modulær
monolitt passer for ditt system



FORVALTBARHET

- Enklere onboarding
 - Utrulling av konsulenter blir enklere
 - Mindre personavhengig
- Lik kode i hele systemet
 - Alle modulene følger samme mønstre og konvensjoner
- Standard teknologi
 - Enklere å finne kompetanse
 - Langsiktige valg

TIDSBESPARELSER

- Hadde tidspress på å bli ferdig
- Tar kortere tid å lage én stor applikasjon enn 10 små
- En del tid spart på CI/CD, release-pipelines o.l.
- Delt kode/rammeverk gjør at hver modul trenger mindre boilerplate
- Mye enklere å onboarde personer i én løsning framfor mange
- Behov for færre eksterne grensesnitt

KRAV TIL SYSTEMET

- Håndterlige krav til ytelse
- Kjente og håndterlige datamengder
- Stabil mengde trafikk (også på lang sikt)
- Mulighet for nedetid (ved behov)
- Ingen kjente behov for å skulle skalere (på lang sikt)
- Korrekthet er viktig

IMØTEKOMME BEHOVENE

- Tett samspill mellom noen av modulene
- Mye felles behov på tvers av modulene
 - Tilgang til kundedata og kjernedata
 - Sporbarhet, GDPR, sikkerhet, logging
- Adresserer kjente utfordringer med dagens løsninger
- Understøtter prosessen med å bytte ut kjernesystem
- Forenkle og gjøre samhandlingsflyter forutsigbare
 - Ingen nettverk = mindre feil
 - Kan faktisk ha en transaksjon som spenner flere moduler*

ERFARINGER MED EN EKTE MODULÆR MONOLITT

(som er på vei ut i produksjon)



MINE REFLEKSJONER

- Iboende kompleksitet blir ikke borte uansett arkitektur
- Tvinger frem en eller annen meldingsutveksling for å beholde løs kobling mellom modulene
- For mindre/enkle systemer er dette unødvendig komplekst
- To grunner til å velge modulær monolitt:
 - Enkelt kunne bryte ut moduler til egne tjenester i fremtiden
 - Håndtere mange ansvarsområder i en enkelt applikasjon
- Du blir å bygge et slags rammeverk for modulene

DET POSITIVE

- Teamet har vært produktivt hele veien
- Få konflikter i versjonskontrollsystemet
- Tror arkitekturen har hjulpet oss å komme i mål med høy kvalitet
- Ikke møtt på hindre som har stoppet oss
 - Har måtte gjøre mindre justeringer i kjernen underveis
- Har klart å oppnå målene
- Yter bra
- Vanskelig å gjøre feil

DET MINDRE POSITIVE

- Unikt oppsett (sannsynligvis samme utfordring for alle)
- Krever noen erfarne utviklere og helst noen med arkitekturerfaring
- Kan bli mange nye teknikker og mønstre
- En del oppstarts-kostnad (men som gir avkastning seg på sikt)
- Finne grensene til en modul er vanskelig
- Kommunikasjon mellom moduler kan være vanskelig å få rett på første forsøk
- Teamet har hatt noen utfordringer med høy minnebruk i IDE

HVA VILLE JEG GJORT ANNERLEDES NESTE GANG?

- Etablert templates/snippets tidlig (mye repetisjon)
- Flere arkitekturtester tidlig (hjelper å gjøre ting riktig)
- Involvert fagsiden tidligere i oppdeling av «bounded contexts»
 - De har ikke nødvendigvis svaret, men løsning kan sparres fram
- Roselyn analyzers for spesifikke arkitekturregler som ikke kan håndheves på annen måte

TAKEAWAYS

- Modularitet er alltid en bra løsning for å lage en godt oppdelt og vedlikeholdbar kodebase
- Om du ikke trenger eller ønsker ekstraarbeidet med mikrotjenester, så er en modulær monolitt et alternativ
- Vær forberedt på å finne opp din egen kjerne som er riktig for ditt system
- Systemet jeg har jobbet på anser jeg som vellykket
- Jeg kommer til å vurdere modulær monolitt neste gang

RESSURSER FOR Å LÆRE MER

- **Distribu-ready with the Modular Monolith**

Layla Porter - NDC Oslo 2024

- Bra foredrag med mange nyttige tips om hvordan bygge en modulær monolitt
- https://youtu.be/L3r1k8_Bi1M

- **Modularizing the Monolith**

Jimmy Bogard - NDC Oslo 2024

- https://youtu.be/fc6_NtD9sol

- **Modular Monolith with DDD** – omfattende demo-prosjekt

- <https://github.com/kgrzybek/modular-monolith-with-ddd>
- Mye dokumentasjon og begrunnelser av valg
- Ikke alt er like bra, men mye å lære

SPØRSMÅL?

Kom gjerne å slå av en prat!

E-post: torstein@enso.no

LinkedIn: [linkedin.com/in/tnicolaysen/](https://www.linkedin.com/in/tnicolaysen/)

Lenke til slides blir lagt ut på meetup.com i løpet av kvelden

