HELSE MIDT-NORGE

IT arkitekturstrategi

Helse Midt-Norge

V. 2.0

Utarbeidet av:	Bård Grødem, HEMIT Rannveig Woll, HEMIT
Godkjent:	Hemit ledergruppe

Dokumenthistorikk			
Versjon	Dato	Utført av	Endringsbeskrivelse
1.0	14.05.2004		Godkjent versjon
2.0	24.04 2006	Bård Grødem Rannveig Woll	Integrasjonsstrategi innarbeidet i arkitekturstrategien (2.6) Lagt til 2.3 Informasjonsmodell, 2.4.1, Virtualisering av driftsmiljø 2.5.2, Meldingsserver 2.5.4 Portal løsninger Mindre endringer (bl.a. IAM, gevinstpotensiale og veien videre)

Innholdsfortegnelse

1 Innledning

1.1 Bakgrunn

Helse Midt-Norge har i sin IT-strategi definert sentrale virksomhetsmål og IT suksessfaktorer. I forbindelse med spesifikasjon av utstyrs- og systemanskaffelser er det definert en teknologistrategi og en it-norm. For det nye universitetssykehuset er det utarbeidet en handlingsplan og en prosjektplan for IT.

- IT-strategi for Helse Midt-Norge, oktober 2002
- Teknologi strategi for Helse Midt-Norge, versjon 1.00 januar 2003
- Handlingsplan IT for det nye universitetssykehuset i Trondheim perioden 2002 – 2006
- Prosjektplan IT for perioden 2002-2006 for det nye universitetssykehuset (plakat)

En fornuftig arkitekturstrategi er en forutsetning for en vellykket implementering av en helhetlig IT-strategi. Mange private og offentlige virksomheter etablerer arkitektur på virksomhetsnivå, dvs. at arkitektur etableres mer helhetlig i virksomheten og ikke innenfor de enkelte applikasjonene som tidligere (monolittisk arkitektur). De viktigste fordelene med dette er flere synergier mellom de ulike applikasjonene (gjenbruk), mindre avhengighet mellom kunde og leverandør og mer kostnadseffektiv og håndterbar integrasjon mellom applikasjoner og tjenester.

Arkitekturstrategien inneholder retningslinjer for hvordan nye systemer skal etableres i Helse Midt-Norge, hvordan de skal benytte standard fellestjenester, hvordan systemer, tjenester og arbeidsprosesser skal dokumenteres og hvordan en felles driftsarkitektur skal se ut.

I likhet med teknologistrategien er dette dokumentet ikke et endelig dokument, men vil kontinuerlig være under revisjon (årlig) for å fange opp utviklingstrekkene i markedet. Samtidig er det heller ikke for detaljert slik at det legger unødvendige begrensninger på leverandørenes eller interne løsningsmuligheter.

1.2 Mål for IT i Helse Midt-Norge

De viktigste IT suksessfaktorene for å kunne nå Helse Midt-Norges virksomhetsmål er:

- IT-systemer som understøtter virksomhetsstyring, økonomistyring og ledelsesinformasjon
- Tilgjengelige, korrekte og oppdaterte pasientdata
- IT-løsninger som underbygger effektiv tjenesteyting
- Felles løsninger for ventetider, kapasitetsinformasjon og booking

1.3 Oppbygning av dokumentet

Leseren av dette dokumentet bør ha noe forhåndskunnskap om arkitektur og integrasjonsteorier. Videre bygger dette arbeidet på det som er gjort i utarbeidelse av Teknologistrategi. Leseren kan med fordel være kjent med disse.

Kapittel 1: sier noe om arbeidet og innhold i arkitekturstrategien.

Kapittel 2: beskriver valgt arkitekturmodell.

Kapittel 3: beskriver hvordan forretnings og kravmodellering gjennomføres i Helse Midt-Norge basert på bruk av Use-Case modellering.

Kapittel 4: beskriver gevinstpotensiale på kort og lang sikt innenfor ulike gevinstområder. .

Kapittel 5: gir en oppsummering og en anbefaling til videre arbeid mht. arkitektur i Helse Midt-Norge.

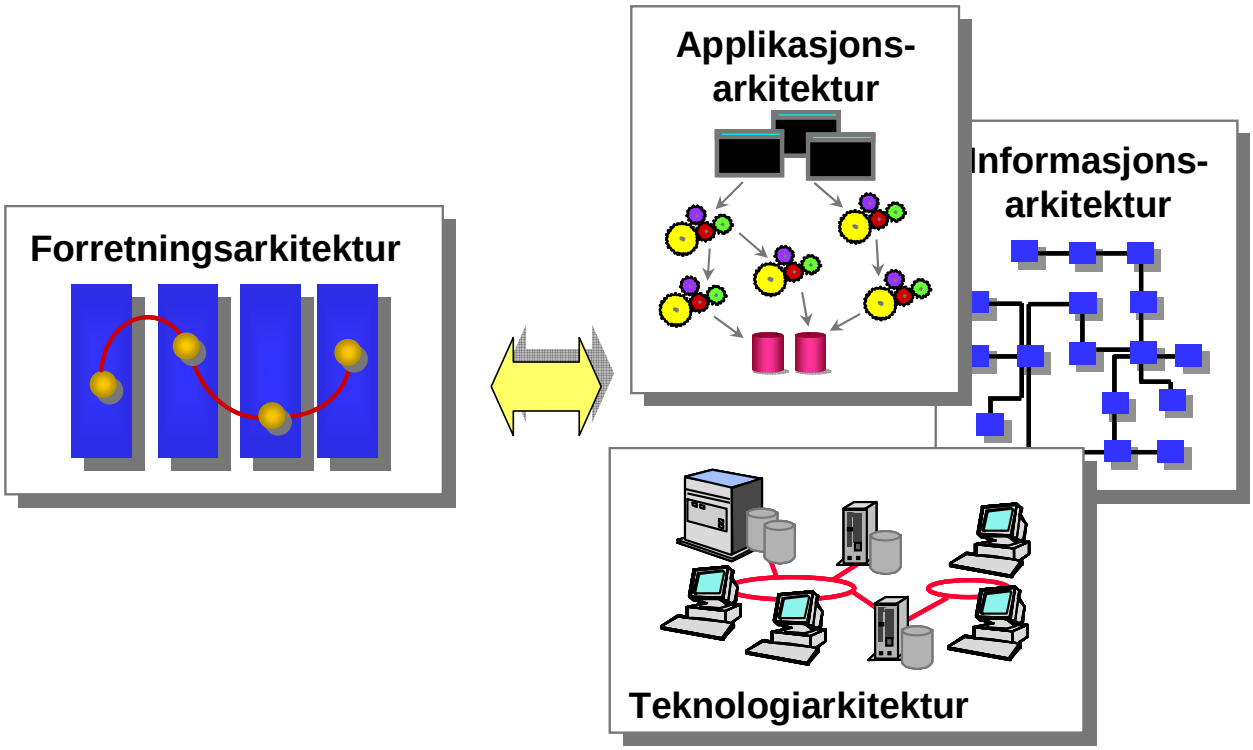
1.4 Målgruppe

Arkitekturstrategien har identifisert ulike interessenter som kan ha en eller annen interesse av denne:

- Ledelse
- Pasient
- Utviklere
- Driftspersonell
- Sikkerhetsansvarlig
- Eksterne leverandører
- Eksterne samarbeidsparter
- Myndigheter

1.5 Virksomhetsarkitektur

Virksomhetsarkitektur (Enterprise Architecture - EA) kan best defineres ut fra dens bestanddeler. Figuren viser disse.



Figur 1 Enterprise Arkitektur

- **Forretningsarkitektur**. Representerer de reelle forretningsprosessene som virksomheten gjennomfører, uavhengig av system- eller implementasjonshensyn. Denne arkitekturen kan f.eks. inneholde forretningsentiteter, -prosesser, -ressurser, o.l.
- **Applikasjonsarkitektur**. Representerer en logisk (teknologi- og implementasjonsnøytral) modell av systemer/tjenester som støtter opp under forretningsarkitekturen. Dette kan være modeller av automatiserte tjenester som støtter opp under forretningsprosessene, avhengigheter mellom de ulike applikasjonene i virksomheten, planer for nye applikasjoner som vil støtte fremtidige behov, o.l.

- **Informasjonsarkitektur**. Representerer modeller som viser hva virksomheten trenger å vite for å kunne gjennomføre prosessene og aktivitetene. Dette kan f.eks. være modeller over forretningsentiteter, deres relasjoner, forretningsregler, o.l.
- **Teknologiarkitektur**. Representerer en modell av maskinvare og programvare som støtter opp under applikasjons- og informasjonsarkitekturen. Dette representerer et mer fysisk syn på virksomhetsarkitekturen.

Applikasjonsarkitektur, Informasjonsarkitektur og Teknologiarkitektur understøtter i modellen Forretningsarkitektur. Vi kan si de understøtter forretningsaktivitetene.

1.6 Avgrensninger

En komplett arkitekturstrategi innbefatter en rekke elementer (se Figur 1 Enterprise Arkitektur).

I utarbeidelsen av arkitekturstrategi for Helse Midt-Norge har prosjektet fokusert på følgende områder:

- Overordnet arkitekturmodell
- Programvarearkitektur
- Applikasjonsintegrasjon
- Driftsarkitektur
- Fellestjenester
- Overordnet sikkerhetsstrategi

I tillegg er en pilot skissert og igangsatt med følgende fokus:

- Programvarearkitektur
- Fellestjenester
- Sikkerhetsmekanismer

Prosjektet har ikke berørt:

- Driftsstrategi
- Teknisk- medisinsk utstyr
- Sikkerhetsstrategi (fullt ut med rollemodell, pki-strategi osv).
- Integrasjon av personell opplysninger

Forretningsmodell, applikasjonsmodell og informasjonsmodell er bare overfladisk berørt. Prosjektet anbefaler at disse elementene berøres i en senere revidering av virksomhetsarkitekturen.

1.7 Suksessfaktorer

1.7.1 Forankring

Forankring innad i Helse Midt-Norge og utad mot leverandører er en vesentlig suksessfaktor.

Forankring bør skje internt mot alle nivåer. Det bør fokuseres på forankring mot det regionale helseforetaket, mot de forskjellige avdelinger i Hemit og mot innkjøpere og beslutningstagere i helseforetakene. Fokus bør være på gevinster og muligheter, men også på avgrensninger og krav. Forankringen bør være tilpasset målgruppen.

1.7.2 Iterativ prosess

Utarbeidelse av en virksomhetsarkitektur er en iterativ prosess. Etter hvert som man anvender arkitekturen vil man gjøre seg erfaringer og man vil erfare at teknologien endrer seg og nye muligheter dukker opp. Virksomhetsarkitekturen bør derfor revideres jevnlig og i henhold til en plan.

1.8 Oppsummering

Helse Midt-Norge har etablert en virksomhetsarkitektur som definerer rammene for etablering av IT-systemer i Helse Midt-Norge. Følgende elementer utgjør pilarene i virksomhetsarkitekturen:

- Tjenestebasert arkitekturmodell
- Driftsarkitektur
- Programvare arkitektur
- Fellestjenester

Den tjenestebaserte arkitekturmodellen er en grunnstein i virksomhetsarkitekturen. Med tjenestebasert arkitekturmodell menes at alle systemer tilbyr sin funksjonalitet via et tjenestesnitt (implementert som en Web Service). Denne tjenesten kan da konsumeres av andre tjenester og systemer. Et eksempel på en tjeneste kan være ”HentPasientData” som henter alle pasientdata fra PAS. Hvordan disse data presenteres er da tjenesten uvedkommende og helt og fullt opp til konsumenten. Det er ønskelig at alle systemer og tjenester bygges opp med n-lags arkitekturmodell som i beskrivelsen av programvare arkitekturen. (Dette er en industristandard arkitekturmodell),

Driftsarkitekturen er designet for å kunne kjøre ALLE systemer som er laget i henhold til virksomhetsarkitekturen. Dette vil forenkle drift og bidra til reduserte kostnader forbundet med maskinvare. I stedet for at alle systemer krever egen maskinvare utvides denne kun ved behov for mer kapasitet. I tillegg kan den også kjøre applikasjoner som ikke er laget i henhold til virksomhetsarkitekturen, men disse vil da måtte defineres som unntak fra enkelte av drifts- og sikkerhetsaspektene ved driftsarkitekturen.

Når systemer tilbyr og konsumerer tjenester er det naturlig å tenke seg at enkelte av disse kan være tjenester som tilbys av Helse Midt-Norge og som benyttes av alle systemer. Det kan være tjenester for brukerhåndtering/pålogging og feillogg/driftslogg.

2 Arkitekturmodell

2.1 Innledning

I forbindelse med utarbeidelse av en arkitekturstrategi for Helse Midt-Norge beskriver vi hvordan framtidige applikasjoner skal bygges opp og integreres i Helse Midt-Norge. Dette vil lette arbeidet med programvareutvikling, integrasjon og innkjøp av programvare. Samtidig skal arkitekturstrategien ivareta at Helse Midt-Norge fortsetter å kjøre eksisterende systemer fra eksterne leverandører.

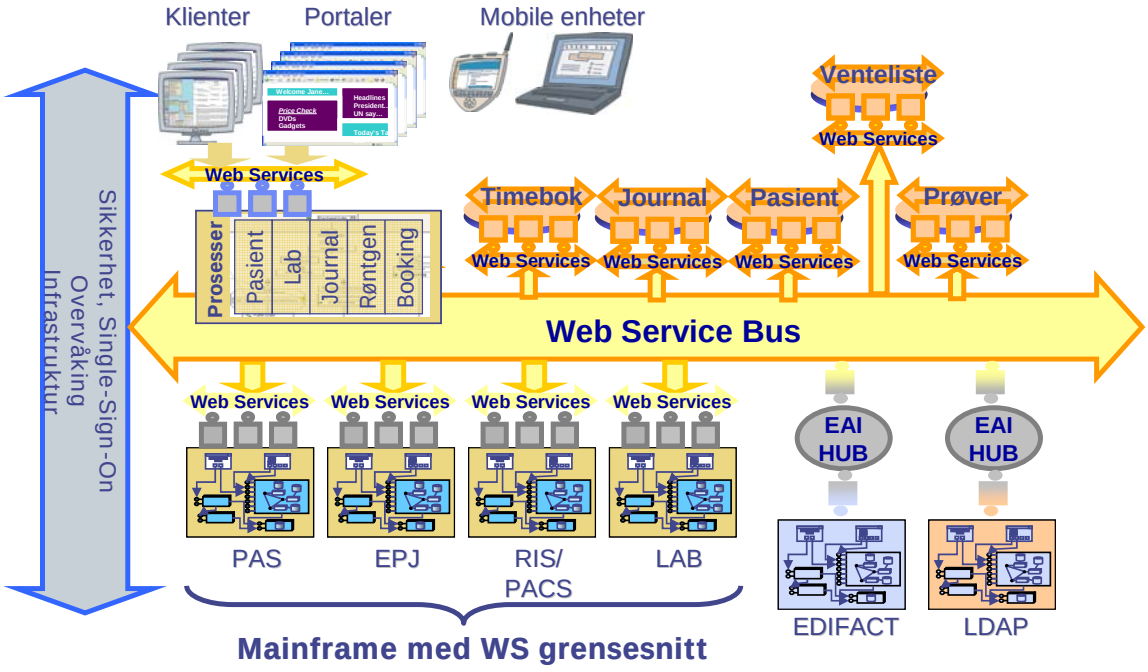
Dette kapittelet beskriver hvordan Helse Midt-Norge ser for seg at framtidens løsninger skal realiseres mht. driftsarkitektur, programvarearkitektur samt bruk av fellestjenester som logging, autentisering, autorisasjon med mer.

Arkitekturmodellen er delt inn i følgende deler:

- Tjenestebasert arkitektur
- Driftsarkitektur
- Programvare arkitektur
- Fellestjenester

I tillegg er sikkerhet og sikkerhetsmekanismer drøftet.

2.2 Tjenestebasert arkitektur



Figur 2 Eksempel på en teknologisk realisering av en tjenestebasert arkitektur

Helse Midt-Norge har valgt en tjenestebasert arkitekturmodell. Alle framtidige løsninger i Helse Midt-Norge skal realiseres i en tjenestebasert arkitektur. En tjenestebasert arkitektur kjennetegnes ved at applikasjoner og systemer tilgjengeliggjør sin funksjonalitet som et sett av tjenester. En applikasjon kan konsumere et sett av tjenester fra forskjellige systemer. Data overføres i XML-format.

Figur 2 Eksempel på en teknologisk realisering av en tjenestebasert arkitektur viser hvordan virksomhetsarkitekturen kan realiseres ved hjelp av en web service bus. Dette konseptet er et anerkjent bilde på en teknisk arkitektur basert på web services.

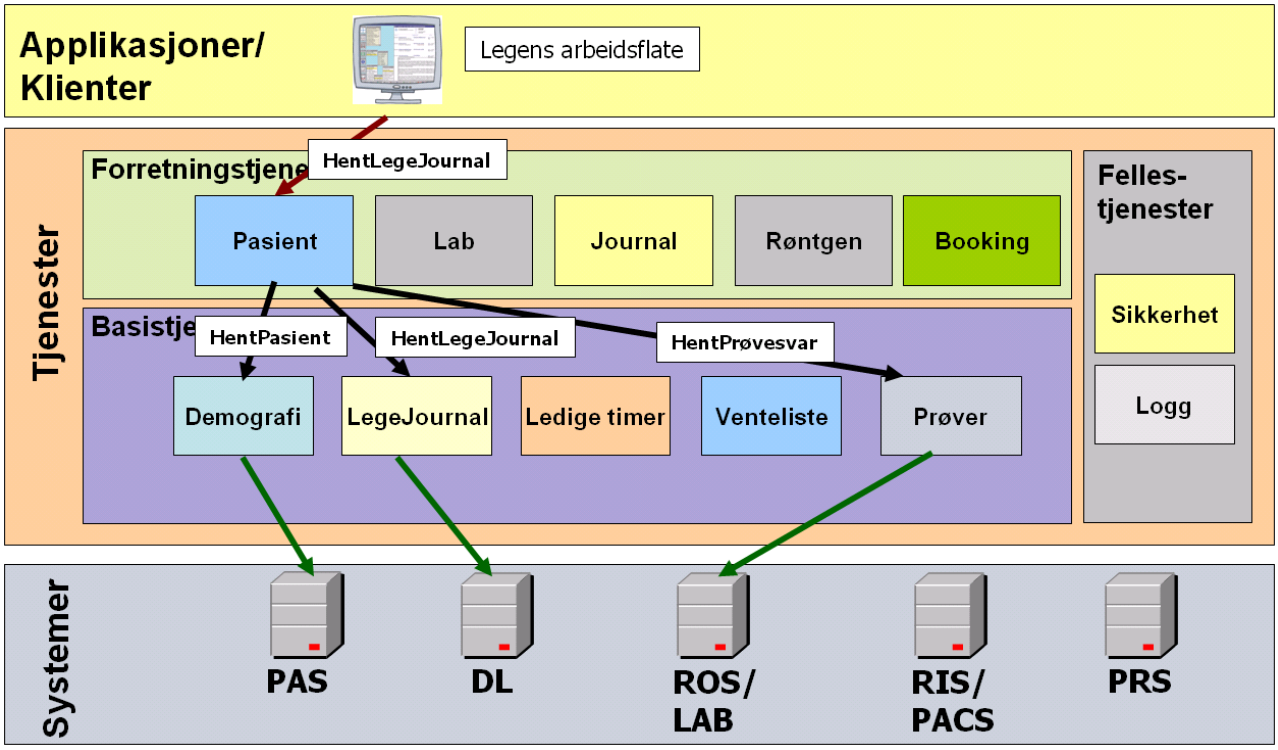
I figuren er Web Service Bus sentral. Web Service Bus er ikke et produkt, men mer en infrastruktur som inneholder kommunikasjonsprotokoller.

I tillegg opererer vi med prosess- og forretningstjenester. Forretningstjenester implementeres i en eller flere tjenester som de ulike typer klienter (brukergrensesnitt) kommuniserer med. Prosesstjenestene kan relateres mer direkte til brukerens arbeidsprosesser. Forretningstjenestene kommuniserer med underliggende basistjenester via web services implementert på ulike plattformer. Dette kan f.eks. være .NET, Java, Mainframes, o.l. Kravet er at disse webservicene følger gjeldende standarder.

Systemer som ikke tilbyr et webservicegrensesnitt direkte kan kapsles inn ved hjelp av en Enterprise Architecture Integration Hub. Dette gjør at tjenester på web service bussen kan kommunisere med tjenester som i utgangspunktet bare er tilgjengelig via andre teknologier uten noe kunnskap om disse teknologiene. Eksempler på når dette er aktuelt, er integrasjon mot EDI baserte meldingssystemer og LDAP baserte katalogtjenester.

.

Helse Midt-Norge har valgt Microsoft .Net som basis teknologi. Web-tjenester som skal implementeres skal støtte standarder for Web-services som implementert i Microsoft Web Service Enhancement 2.0 (og 3.0 når den lanseres).



Figur 3 Eksempel på utnyttelse av tjenester

Figur 3 Eksempel på utnyttelse av tjenester viser hvordan legen sitter med en applikasjon. Denne utnytter en forretningstjeneste som henter journalinformasjon fra flere forskjellige fagsystemer. For legen oppleves en integrert applikasjon som viser sammenstilte data fra de forskjellige underliggende systemer.

2.3 Informasjonsmodell.

For å kunne designe en god klinisk portal, er en avhengig av en god informasjonsmodell. Portal komponentene må vite hvor de skal hente dataene fra og hvor disse skal lagres.

En datamodell beskriver hvor data er lagret og hvem som eier de forskjellige dataelementene og hvem som er ansvarlig for å vedlikeholde dem. (”Master”).

En informasjonsmodell beskriver informasjonsobjekter. Disse objektene inneholder dataelementer, men har ikke noe forhold til hvor de lagres. Objektene kan være sammensatt av dataelementer fra forskjellige systemer.

I Helse Midt-Norge har vi mange it-systemer og det er en omfattende oppgave å lage en slik datamodell. Dette pga. at det finnes mange tusen dataelementer og disse er dublert i mange systemer. Noen av dataelementene som naturlig hører sammen, finnes også i flere forskjellige systemer. Dette vil også komplisere datamodellen.

Vi har derfor sett det lite hensiktsmessig å lage en datamodell som skal dekke alle systemer og som skal kartlegge alle dataelementene i disse systemene. Vi har heller valgt å lage en informasjonsmodell som tar utgangspunkt i de forretningstjenestene som den kliniske portalen (og andre systemer) er avhengig av for å fungere. En informasjonsmodell basert på disse tjenestene vil være uavhengig av underliggende systemer, selv om den vil baserer seg på dataelementer som finnes i de underliggende systemene. Denne modellen vil beskrive informasjonsobjekter som er sammensatt av elementer fra forskjellige systemer. For eksempel vil et pasientobjekt beskrives med elementer både fra det pasientadministrative systemet og journalsystemet. På denne måten får vi en informasjonsmodell som er uavhengig av underliggende systemer, der systemene både kan oppgraderes og byttes ut uten at modellen trenger å endres eller at de tjenestene som benytter modellen trenger å endres.

Et informasjonsobjekt vil typisk kunne realiseres som en eller flere webtjenester som benytter underliggende basistjenester for å lagre dataene.

Denne tilnærmingen vil ikke overflødiggjøre en kartlegging/evaluering av dataelementene i systemene, da den logiske informasjonsmodellen er avhengig av en konsistent struktur på dette i de underliggende systemene. Men det gjør det mye lettere for Helse Midt-Norge å lage og vedlikeholde en offisiell informasjonsmodell som andre aktører skal forholde seg til når de lager løsninger for regionen. Det kan , i første omgang, lages en informasjonsmodell over de mest brukte tjenestene, slik at vi hurtigst mulig får på plass løsninger som er i overensstemmelse med denne modellen. Dette vil ikke minst gjelde for en ny klinisk portal i regionen, men også for andre nye løsninger som skal innføres. Vi bør også ha som målsetting at allerede eksisterende løsninger migreres over i en slik modell på sikt.

2.4 Driftsarkitektur

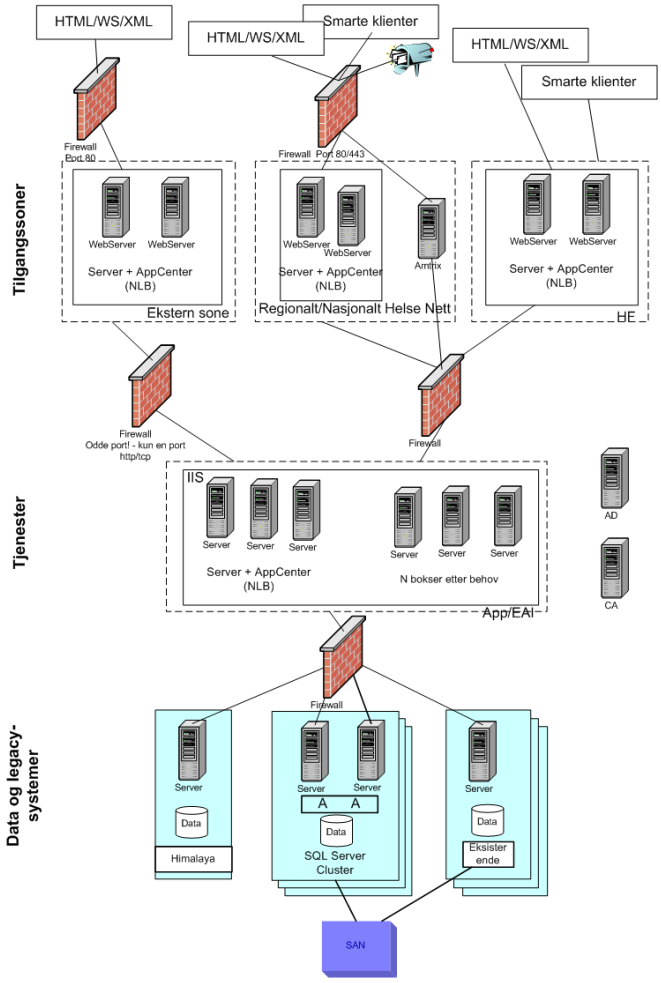
Med driftsarkitektur mener vi det fysiske driftsmiljøet framtidens løsninger i Helse Midt-Norge skal kjøre i. Driftsarkitekturen er designet for å kunne kjøre løsninger basert på en tjenestebasert arkitektur som beskrevet i kap. 2.2 Innledning Tjenestebasert arkitektur

Det er et mål at alle framtidige løsninger kjører i et og samme miljø. Ved behov for mer kapasitet utvides dette istedenfor å etablere driftsmiljø for hvert enkelt system.

Driftsarkitekturen beskriver en lagdelt fysisk arkitektur og klart definerte tilgangssoner. Det etableres egne tilgangssoner for helseforetak, primærhelsetjeneste (Helsenett) og andre. Det etableres egne lag for å kjøre presentasjons-, forretnings- og datatjenester.

Lagene er atskilt vha. brannvegger. Det skal ikke være mulig å gå utenfor disse for å nå underliggende lag. Det skal heller ikke være mulig å gå utenom noen av lagene for å nå underliggende lag.

Driftsarkitekturen er basert på bruk av MS NLB clustering for administrasjon og lastbalansering av presentasjon- og applikasjonslagene, mens database laget benytter Microsoft failover cluster teknologi.



Det skal ikke lagres kontekstinformasjon på presentasjonslaget eller applikasjonslaget. Dvs. at disse komponentene skal kunne gjenbrukes av hvilken som helst klient uten at den er avhengig av å knytte seg til en bestemt sesjon.

Kommunikasjon mellom lagene er basert på bruk av Web Services eller RPC for Com-komponenter.

Driftsarkitekturen implementeres i 4 identiske miljø som benyttes til utvikling, test, produksjonstest og produksjon. Se kap. Leveranse og endringskontroll

2.4.1 Virtualisering av miljøene

I den senere tid har det kommet nye teknikker for å utnytte maskinvaren bedre og få bedre skalering og ytelse ut av anskaffet maskinvare. Denne teknikken kalles virtualisering og kjennetegnes ved at en ikke lengre forholder seg til fysiske maskiner, men et miljø av flere maskiner der en kan definere inn virtuelle servere etter behov. Så lenge det er kapasitet i miljøet, kan en øke antall servere ut over antall fysiske servere i miljøet. Dette gir klare fordeler rundt skalering og ytelse i clustrede miljøer, da en slipper å vente på fysiske installasjoner av maskinvare, men bare trenger å definere opp flere virtuelle servere som skal meldes inn i en clustergruppe. Andre fordeler er at fysisk installasjon av maskinvare blir mer uavhengig av programvareinstallasjoner. Vi kan definere opp virtuelle servere etter behov og supplere med hardware når kapasiteten blir for liten.

Dette understøtter også server konsolideringstankegangen på en utmerket måte. Vi får da samlet flere ressurser i et miljø og på denne måten utnytte ressursene på en mye bedre måte enn med enkelt stående servere der kapasiteten bare delvis utnyttes av systemene.

2.4.2 Tilgangssoner

I tilgangssoner kjører Web-servere med presentasjonstjenester eller andre tjenester som skal nåes fra sluttbruker eller eksterne systemer.

Det etableres tilgangssoner for de regionale helseforetakene, for nasjonalt helsenett og for eksterne aktører. Det vil være forskjellige tjenester innenfor hver tilgangssone, men alle forholder seg til tjenester som tilbys via underliggende lag.

2.4.3 Forretningstjenester

Helse Midt-Norge har valgt å benytte MS.Net som basis i sine løsninger. Tjenester som tilbys skal være basert på .Net (eller Com+ for eksisterende systemer). Tjenester tilbys som Web Services som skal registreres i en katalog/oversikt over tilgjengelige tjenester som aktørene kan forholde seg til for å finne de tjenestene som best dekker det aktuelle systemets behov.

2.4.4 Data og legacy systemer

Tjenestelaget for data og stormaskinsystemer inneholder alle tjenester forbundet med datalagring. Dette er uavhengig av om data lagres i databaser som SQL Server eller Oracle eller om de lagres i systemer som går på stormaskin. Databasene kjører på clustrede løsninger og benytter SAN for fysisk lagring.

2.4.5 Terminalserver

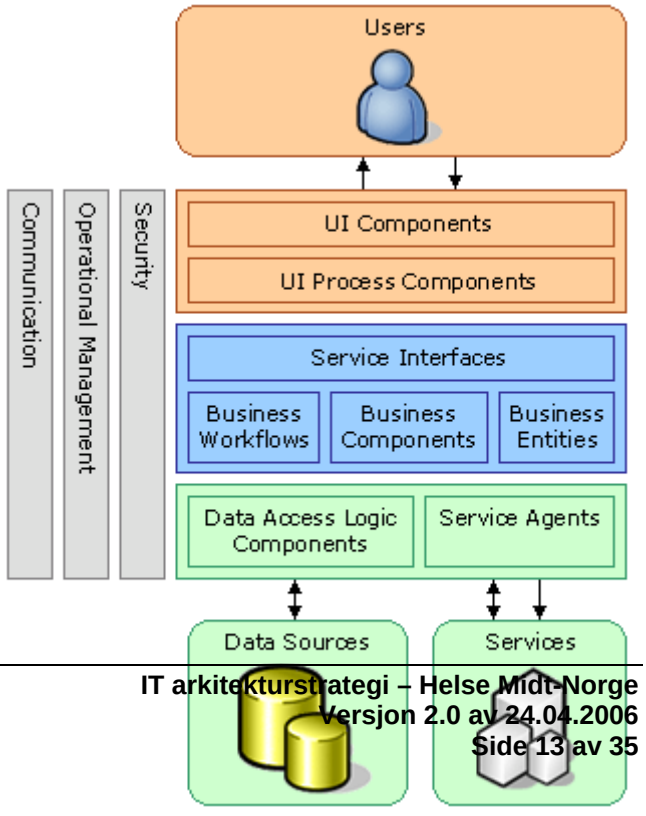
Terminalserver er et aktuelt virkemiddel i Hele Midt-Norge. Terminalserver regnes ikke som en del av driftsarkitekturen, men som et driftsmessig virkemiddel for å forenkle drift og sikkerhet. Dette gjelder for brukere internt i helseforetakene eller for brukere i helsenettet som et alternativ til VPN.

I en overgangsfase, der vi vil ha både tjenesteorienterte løsninger og løsninger med tykke klienter vil terminalservere være berettiget som et virkemiddel for å lette driften av de gamle løsningene.

Alle nye klienter skal kunne kjøres på terminalserver. Terminalserver plasseres i aktuelle tilgangssoner for helseforetak/helsenett.

2.5 Programvare arkitektur

Framtidige løsninger hos Helse Midt-Norge skal kjøre på overnevnte driftsarkitektur basert på tjenestebasert arkitektur. Dette oppnås best ved å etablere en lagdelt applikasjonsstruktur der vi deler i datalag, forretningslag og presentasjonslag. Dette gjøres for bedre å kunne vedlikeholde programvaren, men også for å kunne gjenbruke sentrale deler av programvaren. Mye av koden i en forretningsapplikasjon vil kunne gjenbrukes av andre systemer, bare det er lagt til rette for dette. En av forutsetningene for å få dette til, er da at denne delen av koden blir skilt ut i egne tjenester som deretter blir gjort tilgjengelig for andre systemer. Helse Midt-Norge har valgt Microsoft .Net som basis teknologi.



Presentasjonslaget dekker presentasjonen av data mot brukeren. Dette laget inneholder ingen forretningslogikk, men beskjeftiger seg bare med selve grensesnittet mot brukerne. Dette laget vil ha ansvar for presentasjon på forskjellige typer klienter og andre medier, så som mobiltelefoner, PDA'er, etc.

Presentasjonslaget skal implementeres ved hjelp av .Net.

Forretningslaget vil dekke forretningslogikk og vil inneholde tjenester som dekker de forretningsområdene som presentasjonslaget trenger for å kunne yte riktige tjenester til sluttbrukerne. Laget inneholder forretningsregelverket og presenterer disse på en strukturert måte for sine "konsumenter". Forretningslaget inneholder både prosess- og forretningstjenester. Presentasjonslaget "konsumerer" tjenester fra forretningslaget.

Forretningslaget tilbyr tjenester implementert som .Net komponenter. Disse tilgjengeliggjøres som Web-services.

Alle tjenester skal være dokumentert. Hvis det ikke finnes en mal for dokumentasjon, kan mal for Web-services i Helse Midt-Norge benyttes. Samtidig skal det forefinnes en WSDL for alle tjenester og alle tjenester skal registreres i en katalog. Se også kap. 2.5.4.

Datalaget dekker selve tilgangen til dataene. Dette sørger for at tjenester i forretningslaget får tilgang til de data som den skal manipulere. Dette kan være data fra databaser, stormaskinsystemer eller annet. En tjeneste i forretningslaget vil kun gå mot "sitt" datalag og aldri mot andre tjenesters datalag.

Datalaget implementeres vha. .Net. og ADO.Net.

2.5.1 Cluster

Den valgte driftsarkitekturen legger opp til clustrede løsning med lastbalansering. Dette medfører at alle løsninger må være cluster-aware. Dette innebærer at man spesielt må ta hensyn til følgende:

- Tilstandsløshet eller minst mulig deling av ressurser/tilstand. Dersom tilstand er nødvendig må den være tilgjengelig fra en sentral ressurs som for eksempel en database. Ved clustring av web-applikasjoner må sesjonen ligge i en database eller i ASP.NET sin tilstandstjeneste.
- Sentral logging/monitorering/feilhåndtering. (Se også kap, 1.1.1 Logg).
- Sentral installasjon og konfigurering. (Se også kap. 2.9.3 Leveranse og endringskontroll).

2.5.2 Meldingsserver.

I nåværende arkitektur har vi implementert en meldingsserver som tar hånd om meldinger som skal til/fra primærhelsetjenesten. I en fremtidig arkitektur vil meldingsserveren bli en viktigere komponent i infrastrukturen. Den skal håndtere kommunikasjonen av meldinger mot aktører utenfor det regionale helseforetaket samtidig som den skal håndtere meldinger som sendes mellom systemer internt i det regionale helseforetaket.

Meldingsserveren vil også være en viktig komponent når det gjelder å levere tjenester til en Web Services Bus. Dette kan være tjenester fra systemer som ikke selv kan levere disse som Web Services eller tjenester som man finner mest hensiktmessig gjøres tilgjengelig via meldingsserveren. Meldingsserveren skal kunne konvertere meldinger fra et format til et annet, samt håndtere kryptering, PKI og andre standard formater (slik som EBXML, HL7 etc.). Dette skal kunne gjøres av meldingsserveren selv eller av moduler som er tilknyttet meldingsserveren.

Meldingsserveren skal også kunne kommunisere med fagsystemene via prosedyrekall, web services eller via meldingsutveksling. Helse Midt-Norge har valgt BizTalk fra Microsoft som meldingsserver i sin infrastruktur. Såkalte "plug in" for manglende funksjonalitet i BizTalk kan kjøpes fra andre leverandører.

I dagens løsninger mot eksterne kommunikasjonspartnere (primærhelsetjenesten, kommunale tjenesteleverandører etc.) utveksles alle data via meldingsbasert kommunikasjon. I fremtidige løsninger kan

dette gjøres via web tjenester som sykehusene tilbyr de eksterne systemene. Dette vil knytte disse tettere til sykehus løsningene og derfor tilby disse systemene bedre funksjonalitet enn dagens meldingsbaserte løsninger er i stand til. Dette vil ikke erstatte meldingsbasert kommunikasjon, men være et supplement. Meldingsbasert kommunikasjon vil fremdeles brukes der informasjonen ikke er "tidskritisk", men av natur er av en slik art at den kan oppdateres i det lokale systemet asynkront. Generelt kan alle informasjonselementer som har karakter av å skulle holde et annet system oppdatert, sendes som meldinger. All annen kommunikasjon mellom systemene bør foregå synkront med direkte oppslag via webtjenester utenom meldingsserveren.

2.5.3 Klienter

Arkitekturmodellen understøtter flere klientvarianter. Foretrukket klient vil være Web-basert, implementert i ASP.Net. Det kan i enkelte tilfeller være påkrevd med tykke klienter med et rikere sett av brukergrensesnittkontroller. Tykke klienter skal implementeres som .Net smarte klienter som kan distribueres og kjøres i en Web-leser kontekst.

Følgende tabell viser de alternative klienters egenskaper:

Klient	**Egenskaper**	+	-
Tradisjonell tykk	Tykk klient med standard windows-brukergrensesnitt. Forretningslogikk i klienten. Kommunikasjon mot applikasjon eller databaseserver.	Mulighet for rike brukergrensesnitt Kan integreres med medisinteknisk utstyr/periferienheter	Krever lokal installasjon og konfigurering Applikasjonslogikk lokalt Databasetrafikk til klient
Web klient	Tynn klient med web-grensesnitt. Forretningslogikk ligger på server.	Ingen lokal installasjon og konfigurering	Mindre rike brukergrensesnitt Ikke medisinteknisk utstyr/periferienheter
Smart .Net Klient	Tynn klient med windows-brukergrensesnitt. Ingen forretningslogikk i klienten. Kommunikasjon mot applikasjonsserver over webservices. Kan distribueres vha. web browser og kjører i samme kontekst som denne. Dette er den foretrukne klient typen i Helse Midt-Norge.	Ingen lokal installasjon og konfigurering Mulighet for rike brukergrensesnitt Kan integreres med medisinteknisk utstyr/periferienheter Muliggjør høy sikkerhet (sandkasse)	
Terminalserverklient	Standard tykke, smarte eller web klienter som kjører vha. terminalserver.	Ingen lokal installasjon og konfigurering ut over terminalklienten	Ikke medisinteknisk utstyr/periferienheter

Håndholdt klient	Pda eller TabletPC klient. Kjører tykke (smarte) klienter med kommunikasjon med applikasjonsserver.		Mindre rike brukergrensesnitt Ikke medisinteknisk utstyr/periferienheter

2.5.4 Portal løsninger.

De fleste applikasjoner i Helse Midt-Norge er i dag av en type som er designet for å løse behov innenfor et bestemt medisinsk eller administrativt område. De er relativt lite integrert med hverandre og tvinger derfor brukerne til å skifte mellom disse systemene for å få dekket sitt informasjonsbehov. Det er riktignok laget en del integrasjoner mellom de store systemene for å bøte på de verste manglene, men fremdeles er det et stort behov for løsninger som baseres på arbeidsprosessene som utføres heller enn den inndelingen i fagområder som dagens løsninger tilbyr. Dagens løsninger tilbyr allerede en god del tjenester som kan utnyttes i en fremtidig portal løsning. Det er derfor viktig at all nye systemer som anskaffes også blir levert som tjenester, slik at disse også kan inngå i en portalløsning. Det samme gjelder også for ny funksjonalitet som leveres på de eksisterende løsningene i Helse Midt-Norge, samt at allerede eksisterende tjenester levert på en annen plattform enn webtjenester blir konvertert til webtjenester ved første anledning.

Portal løsningen skal være basert på allerede kartlagte arbeidsprosesser og skal tilby brukerne et verktøy (portal) som er tilpasset deres arbeidsrutiner uavhengig av de underliggende systemer. Portalen skal understøtte arbeidsprosessene ved å benytte de enkelte underliggende systemers tjenester og sammenstille disse i en arbeidsflate som er tilpasset den enkelte bruker.

For å få til et enhetlig brukergrensesnitt vil vi, i noen tilfeller, lage et eget brukergrensesnitt i portalen. For løsninger der grensesnittet er velegnet forvisning i en portal, vil vi kunne implementere eksisterende brukergrensesnitt i portalen som kan inneholde andre systemers funksjonalitet og oppbygning, men som har sin egen layout og design . Dette vil sikre oss en god kommunikasjon mellom de enkelte partene i portal løsningen, samt bidra til et funksjonelt og enhetlig grensesnitt mot brukerne. Om portalen skal baseres på web grensesnitt eller smarte klienter, må utredes nærmere og være en del av et forprosjekt som kartlegger rammeverket for portalløsningen. Rammeverket skal dekke, foruten tekniske krav, også krav knyttet til funksjonalitet, drift og vedlikehold. Rammeverket skal også kunne sørge for god og sømløs kommunikasjon mellom de forskjellige modulene i portalen. Disse forretningsmodulene bør da bygges som såkalte smartparts som kan flagge hendelser som oppstår. Disse hendelsene kan andre abonnere på , slik at de endrer sine grensesnitt i takt med de andre komponentene i portalen. Dette vil sikre oss at sammenhengen i informasjonen blir ivaretatt på en mye bedre måte enn dagens integrasjonsløsninger kan tilby.

Portal løsningen vil være et viktig bidrag i å skaffe helseregionen it-løsninger som er tidsbesparende, kvalitetshevende og som også vil bidra til en mer papirløs hverdag på sykehusene.

2.6 Integrasjon

2.6.1 Realisering

Integrasjon mellom komponenter/systemer skal skje ved bruk av Web Service teknologien. Denne teknologien er tilpasset en tjenestebasert arkitektur og gjør det mulig å kommunisere over flere ulike plattformer. Alle tjenester som implementeres i virksomheten skal implementeres som Web Services i henhold til gjeldende standarder. Alle tjenestekomponenter skal ha ansvar for egne data, og integrasjonspartene skal være enig om kontrakten (grensesittet).

I utgangspunktet skal tjenestekomponenten være ansvarlig for autentisering/autorisasjon av avsender, logging/trace/overvåkning, men det skal i størst mulig grad brukes mekanismer som tilbyes på

virksomhetsnivå (f.eks. sikkerhetstjener, loggtjener, overvåkning, m.m.). Tjenestekomponentene skal også forholde seg til gjeldene infrastruktur og driftsarkitektur.

2.6.2 Andre integrasjonstyper

Det er et mål å integrere tjenester uavhengig av hvilken plattform de er realisert på. Teknologiske føringer er derfor styrt av dette, og arkitekturstrategien legger til grunn en virksomhetsarkitektur basert på åpne Web service standarder. Det kan likevel være behov for andre typer integrasjoner. Nedenfor er en tabell over ulike integrasjonsmåter og krav som stilles ved bruk av dem. Ved integrasjon av systemer/komponenter er det viktig å vurdere hvilket transaksjonsbehov man har.

Krav / Type	Kommunikasjon	Forståelse	Samhandling	Eksterne
Generelt	SOAP	Dokumentasjon WSDL	Sikkerhet Tilgjengelighet Ytelse Konfigurasjon	Overvåkning Stabilitet Ytelse Distribusjon Konfigurasjon Myndighetskrav
Transaksjon	Synkron SOAP HTTP	WSDL Tilstandsløs	WS-Transactions WS-Security WS-Trust Ytelse	Eksisterende infrastruktur WS Grensesnitt skal godkjennes av HMN
Melding	Asynkron SOAP HTTP HMN's køteknologi	XML/EDIFACT WSDL/XML Skjema Standard meldingsformat (KITH) Tilstandsløs	WS-Transactions Stabil kølengde Ressursforbruk	Eksisterende infrastruktur WS Grensesnitt skal godkjennes av HMN
Replikering	Ingen spesielle	Dokumentert informasjonsmodell	Ytelse Lengde Ressursforbruk Avhengigheter Minst mulig låsing	Eksisterende infrastruktur Eier er ansvarlig for systemet og data

Tabell 1. *Krav ved bruk av de ulike integrasjonstypene.*

2.6.3 Krav til kommunikasjon

Kommunikasjonsbehovene til de forskjellige integrasjonstypene er noe forskjellige. For de transaksjons- og meldingsbaserte typene skal kommunikasjonen skje over Web service bussen. En definisjon av Web service bussen i denne sammenheng er en mekanisme som gjør det mulig å utveksle SOAP meldinger over f.eks. HTTP via ett eller annet fysisk medium som for eksempel Ethernet, WLAN og GSM.

For den replikeringsbaserte typen er valget til kommunikasjon fritt, men skal i størst mulig grad være forenlig med den infrastrukturen og driftsarkitekturen som finnes.

Fra eksterne samarbeidspartnere som for eksempel legekontorer, apotek, rikstrygdeverket, forskning og undervisning gjelder de samme krav som ellers, dvs. at kommunikasjonskravene er avhengig av integrasjonstype.

2.6.4 Krav til forståelse

Dokumentasjonen skal være så formell[1] som mulig, og i tilfeller hvor integrasjonen skal skje over Web service bussen skal dokumentasjonen være formelt dokumentert ved hjelp av WSDL. I tillegg skal integrasjonen være dokumentert slik at de som skal benytte seg av den har tilstrekkelig og korrekt informasjon. Dette omfatter blant annet:

- Metodekall m/argumenter inn og returverdier
- Mulige feilsituasjoner og hvordan disse signaliseres og bør håndteres
- Ev. ikke opplagte konsekvenser av integrasjonen

Dokumentasjon skal være relatert til informasjonsmodellen for HMN. Dette gjelder spesielt for replikeringsbasert integrasjon hvor dokumentasjonen skal beskrive hvilke data som berøres, men også for de andre typene hvor datastrukturene/entitetene som utveksles skal være i samsvar med informasjonsmodellen.

Følgende er eksempler på hva dokumentasjonen kan bestå av:

- Tekstlig beskrivelse (Word dokument, PDF, HTML, etc.)
- Hjelp som kan integreres i Visual Studio .NET

Eksempelapplikasjon/demo

2.6.5 Krav til samhandling

Noen punkter som må være oppfylt for å ha en optimal samhandling:

- Sikkerhet
  - Felles brukerrepresentasjon
  - Felles autentiseringsmekanisme (single-sign-on)
  - Felles autorisasjonsmekanisme
  - Auditing (historikk/sporing - krever felles brukerrepresentasjon)
  - Meldingsintegritet
  - Meldingskonfidensialitet
- Tilgjengelighet
  - Oppetid på høyde med tjenesten man integrerer seg mot
  - Ingen single-point-of-failure
  - Felles mekanisme for overvåking og logging
- Ytelse[2]
  - Skalerbarhet på høyde med tjenesten man integrerer seg mot
  - Ressursbehov som er så lavt at det ikke påvirker andre komponenter
- Konfigurasjons- og versjonsstyring for å sikre at oppdatering av systemer og komponenter ikke har ødeleggende konsekvenser for andre deler av systemet.

Integrasjon må være basert på tillit mellom integrasjonspartene, men mot eksterne systemer og i noen grad systemer levert av eksterne leverandører må det finnes avtaler som plasserer ansvar.

---

[1] Det tenkes her på at dokumentasjonen bør følge en definert struktur. WSDL ser en slik definert struktur som på grunn av sin utvetydighet kan leses av verktøy.

[2] Disse kravene er ikke like viktig for replikeringsbasert integrasjon.

2.6.6 Andre eksterne krav

Generelle krav for administrasjon, drift og vedlikehold av infrastrukturen:

- Overvåkning: Feilmeldinger og ytelsesmålinger[3] skal sendes til drifts felles overvåkingssystem[4]
- Stabilitet: Stabiliteten på integrasjonsmekanismen skal ikke være dårligere enn systemene som integreres
- Ytelse: Ytelsen på integrasjonsmekanismen skal ikke være dårligere enn systemene som integreres
- Distribusjon av programvare: Distribusjon av integrasjonsmekanisme skal følge de rutiner som finnes for systemene forøvrig
- Konfigurasjon: Realiseringen av en integrasjon må kunne konfigureres til å kunne kjøres i forskjellige programvaremiljøer[5], og den må kunne enkelt tilpasses endringer i infrastrukturen. Eksempler her er:
  - IP/URL til Web Service/meldingskø
  - Om trace er av eller på
  - Valg av kommunikasjonsprotokoll (for eksempel om https skal brukes istedenfor http)

Myndighetene ved Datatilsynet har en del generelle krav når det gjelder personvern. Angående sikkerhet kan følgende punkter nevnes, men dette må ikke ses på som en komplett liste:

- Valg av integrasjonsmåte må gjøres i henhold til lovverket
- Historikk (journal)
- Arkivering/langtidslagring

Pasienten er en viktig interessent for all IT, men sett fra et integrasjonsperspektiv er følgende viktig:

- Informasjon til rett person til rett tid

Kvalitet på informasjon

2.7 Sikkerhetsarkitektur

2.7.1 Innledning

En sikkerhetsarkitektur skal gjenspeile hva bedriften legger vekt på når det gjelder autorisasjon og autentisering. Den skal inneholde opplysninger om hver enkelt bruker, hver enkelt tjeneste og tilgangen til disse. Sikkerhetsarkitekturen vil inneholde tjenester for pålogging, definering av roller, definering av tilgang til tjenester, sertifikathåndtering etc.

I dagens situasjon er det slik at enhver applikasjon vedlikeholder sin egen sikkerhetsstruktur med brukernavn/passord, roller og autorisasjon. Brukerne logger seg på nettet med brukernavn/passord og må deretter logge seg på alle de applikasjoner som vedkommende skal bruke. Dette medfører at de må huske på en hel rekke brukernavn/passord. Dette er en stor risikofaktor med henblikk på sikkerhet og for brukerne oppleves dette som tungvindt. For administratorene medfører dette et stort vedlikehold for å holde applikasjonene à jour med brukere/passord og roller. Ved å etablere en sentral sikkerhetstjener vil Helse Midt-Norge få en sikkerhetsinfrastruktur som håndterer elementer som autentisering, autorisasjon, Single Sign On (SSO) og rollemodeller.

[3] Tenker her på profiling/statistikk.

[4] For eksempel HP OpenView

[5] For eksempel utvikling, test og produksjon.

I dag benyttes Active Directory (AD) for brukerautentisering i Helse Midt-Norge. Det er naturlig at en sikkerhetstjener hos Helse Midt-Norge bygger på AD.

2.7.2 Roller, autentisering og autorisasjon

For autentisering skal Active Directory (AD) benyttes. Ved å kombinere AD med Authorization Manager (AM), som er et tillegg til AD for rollestyring, vil en kunne etablere en kraftig mekanisme for brukerautorisasjon og rollestyring.

Autentisering skal gjøres ved å benytte integrert sikkerhet med AD. Bruker logger seg på mot AD, og skal ikke møte flere spørsmål om brukernavn og passord. Alle applikasjoner skal derimot benytte informasjon om pålogget bruker for å identifisere og verifisere brukerens tilgang.

En tjenestebasert arkitekturmodell ligger vel til rette for rollebasert autentisering. Det bygges opp en rollemodell som defineres mot AD og beskriver hvilke operasjoner de forskjellige rollene skal kunne utføre. Roller kobles mot brukere eller brukergrupper i AD. Verifikasjon av tilgang til applikasjoner, og funksjoner i applikasjonene kan gjøres mot AD, men vil i overskuelig fremtid være avhengig av det enkelte system sin rollemodell og tilgangsstyring.

Ved å kombinere autorisasjon mot AD og autentisering mot AM i alle lag i en tjenestebasert arkitektur oppnår vi sikkerhet i flere lag før bruker når datatjenestene. Dette er et bærende prinsipp i Helse Midt-Norges arkitekturmodell.

Arkitekturmodellen muliggjør aksess fra brukere utenfor helseforetak og helseregion. Sikkerhetsregleverket i AD og AM kan spesifisere eksakt hvordan eksterne henvendelser skal håndteres. Dette inkluderer hva de skal få lov til å gjøre og hvilke data de har tilgang til. Dette kan være primærhelsetjeneste, pasienter, pårørende, myndigheter etc. Disse vil autentiseres og deretter få tilgang til spesielle tjenester som er autorisert for en begrenset del av dataene.

Det vil bli en utfordring å løse et behov for at mange brukere på en dag skal kunne benytte samme arbeidsstasjon for å søke etter eller registrere informasjon angående pasienter. Dette kan gjelde arbeidsstasjoner i miljøer der det er trangt om plassen, døgnkontinuerlig drift med mange ansatte som rullerer på arbeidsoppgavene og det er behov for rask tilgang til pasientinformasjon. Det kan ta for lang tid for den enkelte ansatte hvis autentisering må skje ved nettpålogging på arbeidsstasjonen for at han/hun skal kunne benytte en applikasjon for å søke fram ett enkelt blodprøvesvar for en pasient.

Sikkerhetsmessig er anonyme PC'er i nettet som flere brukere benytter i fellesskap en sikkerhetsrisiko. Sikkerhetsarkitekturen legger derfor til grunn at alle brukere skal autentisere seg med pålogging. Evt. problemer forbundet med tid til pålogging og lasting av profil må løses.

2.7.3 Single Sign On

Et mål med arkitekturmodellen er innkapsling og tilgjengeliggjøring av "gamle" applikasjoner i nye løsninger. Et flertall av disse har egne brukerdatabaser med brukerinformasjon som brukernavn og passord.

For å få til en lettere tilgang til applikasjonene, vil det bli innført en tjeneste for Single Sign On (SSO) som en del av sikkerhetsarkitekturen. Denne tilbyr et kryptert lager for brukerdata koblet opp mot pålogget bruker i AD. Ved bruk vil SSO-tjenesten hente ut brukerdata, dekryptere og autentisere brukeren automatisk.

Innføring av SSO vil lette dagens situasjon med mange brukernavn og passord som brukerne må forholde seg til. Dette vil gi økt sikkerhet i alle ledd, da brukerne bare har ett brukernavn og passord å forholde seg til. Innføring av SSO vil også føre til reduserte driftskostnader i forhold til reduserte henvendelser til brukerhjelp/helpdesk om glemte/utgåtte passord. Løsningen er nå etablert for en del systemer i helseregionen og vil utvides etter hvert til å dekke alle relevante systemer. Vi vil også legge føringer på at nye system som anskaffes, skal ha innebygget integrert pålogging/sikkerhet.

2.7.4 Integrering med personalsystem

Opplysningene som danner kjernen i en sikkerhetstjeneste bør komme fra et system som vedlikeholder brukeropplysninger og som naturlig oppdateres når det skjer endringer på brukernes rettigheter. Det være seg at de slutter, endrer stilling, skifter avdeling etc. Et slikt system kan være personalsystemet eller et annet system der brukerne må registres når de begynner og slettes når de slutter. Disse opplysningene kan da overføres automatisk til sikkerhetstjeneren som legger opp en basal tilgangstruktur for brukerne. Spesielle tilganger må da registreres i ettertid. Dette etableres nå som en del av SSO tjenesten i Helse Midt-Norge.

Dette inngår i konseptet rundt IAM (Identity and Access Management) for Helse Midt-Norge. Konseptet bygger på at brukeridentiteten oppstår i personalsystemet (som er det første systemet brukerne møter) herfra sprees identiteten ut til AD og SSO systemet, der det lages en profil for brukeren (hjemmeområde etc.) I SSO lages det tilgangskjema til applikasjoner, slik at avdelingsleder og andre med personalansvar kan gi brukere tilgang til systemer de skal bruke i sin tjeneste på sykehusene. SSO sørger da for å registrere brukere og passord i systemene og lagrer dette for senere bruk mot disse systemene. SSO håndterer også regelmessig bytte av passord som enkelte system forlanger. Når bruker så starter et system, vil SSO enten bruke Windows påloggingsidentiteten direkte mot systemet eller hente brukernavn/passord fra SSO databasen for system og gjeldende bruker og logge brukeren direkte inn i systemet. Videre inn i systemet er det systemets egen sikkerhetsprofil som styrer tilgang til data og funksjonalitet.

Når en bruker slutter, sørger IAM for at identiteten blir slettet/får utgått dato i de respektive systemene. Dette skjer også via samhandling med personalsystemet.

2.7.5 Kryptering og bruk av PKI

Arkitekturmodellen åpner opp for andre former for både autorisasjon og autentisering enn brukernavn/passord. Dette kan være løsninger rundt smartkort, biometrisk pålogging og digitale sertifikater. Modellen legger også til rette for bruk av PKI for kryptering av informasjon fra brukere og mellom bestanddeler i en løsning.

Helse Midt-Norge har valg å basere seg på teknologi fra Microsoft. Microsoft tilbyr løsninger for PKI gjennom Windows 2000/2003 Server integrert med AD. Samtidig vil PKI være en integrert del av meldingsserveren og vil her håndtere kryptering og sertifikater for alle kommunikasjonspartnere som går gjennom meldingsserveren.

2.8 Fellestjenester

I en arkitekturmodell der forskjellige systemer tilbyr et sett av tjenester er det også naturlig at en del tjenester er felles og tilbys alle systemer. Slike tjenester kan deles i kliniske tjenester og infrastrukturtjenester. Eksempler:

Kliniske tjenester

- Pasientinformasjon
- Reservasjon mot innsyn (eks. VIP-pasienter)
- Ventelistehåndtering
- Timebestilling
- Kodeverk
- Organisasjonsinformasjon

Infrastrukturtjenester

- Sikkerhetsfunksjoner slik som autorisasjon og autentisering.
- Aktivitetslogging i form av tjenester som kan spore hva som er gjort, av hvem og når.
- Tracing. Dvs. tjenester som kan kalles opp for å utføre feilsøking på flere valgfrie nivå. Denne tjenesten må være slik at den kan nåes mens systemet er i full drift.
- Kommunikasjon. Tjenestene skal ikke bry seg med hvordan kommunikasjonen foregår mellom de forskjellige lagene. Dette skal ligge som et grunnlag som tjenestene skal benytte seg av. Denne kommunikasjonen er typisk SOAP over TCP/IP.

Tjenestene ligger som en basis i arkitekturmodellen og skal kunne benyttes av alle systemer. Følgende tjenester etableres som en del av Piloten:

- Autentisering og autorisasjon
- Single Sign-On (SSO)
- Logg (systemlogg/hendelseslogg)

2.8.1 SSO, autentisering og autorisasjon

Det vil bli etablert et felles rammeverk/tjeneste for autentisering, autorisasjon og single-sign on. Tjenestene vil bli implementert på toppen av AD/AM.. Tjenestene vil bli implementert vha. sikkerhetsmekanismer i Web Services Enhancement 3.0 (WSE 3.0).

2.8.2 Logg

Det etableres en felles logg- og overvåkningstjeneste. Dette for å sikre enklere og mer enhetlig overvåkning av framtidige systemer. Alle systemer skal logge til Web-tjenesten eller logge mot standard Windows Event Log. Alle systemene skal angi en unik EventSource og

Tjenesten vil bli implementer som en Web-service over Windows Event Log. Tjenesten vil bli integrert med et nettovervåkningssystem for konsolidering av logginfo og overvåkning/rapportering.

Første versjon av tjenesten vil bli spesifisert og implementert første halvår 2008.

2.9 Lokalisering av tjenester

Lokalisering, oppslag på navn og versjoner av tjenester må foregå på en kontrollerbar måte. Standard måte å gjøre dette på er å sette opp en katalog-tjeneste. Basis for katalogen er en WSDL som benyttes for beskrive en Web-service, hva den gjør og hvilket grensesnitt den tilbyr. Alle tjenester som tilbys i Helse Midt-Norge skal ha en WSDL.

Alle tjenester som tilbys i Helse Midt-Norge kan publiseres i en katalog over tilgjengelige tjenester. Alle systemer som benytter en tjeneste kan finne lokalisering av denne i Helse Midt-Norges tjenestekatalog.

I tillegg vil WS-Policy benyttes for å regulere krav til kommunikasjon, sikkerhet med mer, mot en tjeneste.

2.10 Kvalitetskontroll

Arkitektur og infrastrukturmodellen legger opp til gjenbruk av tjenester og infrastruktur. I en slik kontekst vil kvalitetsrutiner rundt utvikling og utrulling og drift være sentral. Med infrastrukturmodell menes her struktur rundt serverkonsolidering og gjenbruk av servere.

2.10.1 Krav til dokumentasjon

Alle leveranser av tjenester, klienter og annen programvare til Helse Midt-Norge skal dokumenteres ihht. vedtatte standarder. Dokumentasjonen skal leveres sammen med programvaren og være en del av

leveransen. Leveransen er ikke å betrakte som komplett før dokumentasjonen er komplett og levert ihht. dokumentstandarden.

Dokumentasjonen skal være så detaljert at Helse Midt-Norge selv kan stå for drift og vedlikehold av løsningene. Herunder vedlikehold av databaser og tjenester knyttet til løsningen.

Der det er levert kildekode med løsningen, skal denne dokumenteres slik at Helse Midt-Norge selv kan vedlikeholde denne. Der hvor dette ikke er tilfelle skal løsningen dokumenteres slik at alle sammenhenger mellom tjenester og hva hver enkelt tjeneste gjør, kommer klart frem.

Alle grensesnitt skal dokumenteres slik at det går klart frem hvilken informasjon som flyter mellom de forskjellige grensesnittene og hvilken semantikk som ligger i grensesnittene.

Definisjonene av tjenestene skal dokumenteres med hjelp av WSDL.

I dokumentasjonen skal det klart fremgå hvilken type klienter som er i bruk og som støttes. Forskjellige klienttyper vil ha forskjellige behov for dokumentasjon, og det skal foreligge egen dokumentasjon for hver type klient som støttes.

Alle feilkoder og feilsituasjoner skal beskrives, slik at det er lett å finne forklaring på hendelser i programvaren. Feilkodene skal ledsages av en oppskrift på hvordan feilsituasjonene løses og hvordan en kommer tilbake til normalsituasjonen igjen.

Leverandørene skal oppgi størrelsene på meldinger som går inn/ut av sine systemer. Disse bør av kapasitets og ytelsesgrunner holdes på et minimum. Det aksepteres ikke at hele objektmodeller sendes fra systemene, da disse kan være store og derfor redusere kapasitet og ytelse i systemet, på server og i nettverket.

En komplett dokumentasjon i følgende deler:

Systemdokumentasjon - Skal gi en komplett oversikt over systemet og sammenheng mellom tjenestene i systemet. Disse skal beskrives slik at slik at det går klart frem hvilken rolle de har i systemet og hvilke handlinger de utfører. Dokumentasjonen skal også vise hvilken informasjon systemet produserer og tar imot.

WSDL beskrivelsene skal være en del av systemdokumentasjonen og dokumenteres i henhold til standarden for WSDL i Helse Midt-Norge.

Brukerdokumentasjon - Skal gi brukerne en innføring i bruk av systemet og beskrive alle funksjoner slik at brukerne er i stand til å operere systemet uten videre opplæring. Dokumentasjonen kan gjerne være elektronisk og en del av systemet og være implementert som Windows Help filer.

Driftsdokumentasjon - Skal gi en oversikt over hvilke driftsprosedyrer som skal gjennomføres for å sikre et stabilt system med avtalt oppetid og tilgjengelighet. Hvis systemet tilbyr batchorienterte rutiner, skal disse også beskrives her.

Installasjonsdokumentasjon - Dokumentasjon som forteller hvordan systemet skal installeres og krav som må være oppfylt før installasjonen finner sted.

Dokumentasjonen skal være så detaljert at systempersonell skal kunne sette opp systemet uten assistanse fra leverandør.

Under installasjon skal det genereres en installasjonslogg som dokumenterer hva som er installert, på hvilken måte og hvordan hele installasjonen foreløp, herunder også tidsforløp. Alle uregelmessigheter skal dokumenteres og det skal angies hvilke tiltak som ble iverksatt for å løse disse.

Endringsdokumentasjon - Her skal det dokumenteres hvilke endringer som er utført på systemet. Dette kan være både maskinvare endringer og programvare endringer.

Det skal dokumenteres både endringer på basis programvare og endringer på systemet som er levert. Endringsdokumentasjonen skal inneholde tid, hvem som utførte endringen og en beskrivelse av eventuelle problemer rundt endringen. Alle uregelmessigheter skal dokumenteres og det skal angis hvilke tiltak som ble iverksatt for å løse disse.

Helse Midt-Norges maler for dokumentasjon skal benyttes der denne finnes.

2.10.2 Trace (sporbarhet)

For lett å kunne finne feil og mangler i løsninger, må all aktivitet som foregår mellom de forskjellige tjenestene i arkitekturen kunne spores. Det er derfor nødvendig at alle tjenester skal ha innebygde mekanismer som tillater tracing på forskjellige nivåer.

I dagens programportefølje er det bare et fåtall løsninger som har innebygd støtte for tracing. Når det meldes feil i en løsning, er det en meget tidkrevende prosess å finne årsaken til feilen, spesielt når flere systemer er integrert. Det blir lett en lang dialog mot leverandører og ressurspersoner for å kunne rette på forholdet. I mange tilfeller fører dette også til uønsket nedetid. På samme måte er det problematisk å finne årsaken til responstidsproblemer. Det er meget vanskelig å kunne si hvor flaskehalser oppstår i et komplekst datasystem. Dette alene er årsak til mye frustrasjon ute hos brukerne.

Trace er et effektivt hjelpemiddel for feilsøking og for å finne flaskehalser i en løsning. Alle nye løsninger skal kunne traces. Trace skal kunne slås av og på uten restart av løsningen. Minimum krav til hva som skal rapporteres er en tjenesteid som beskriver hvilken tjeneste det kjøres trace på, og som kan brukes til å identifisere tjenesten gjennom alle tjenestelagene. I tillegg må det rapporteres alle vesentlige parametre som tjenesten bruker, så som sentrale variable og andre tjenester som denne benytter seg av. Alle tjenester skal også rapportere tidsvariable, som angir tidsforbruk i de enkelte delene av tjenesten. Dette kan brukes til å finne flaskehalser i applikasjonene.

2.10.3 Leveranse og endringskontroll

Når løsninger og tjenester gjenbrukes og kjører i et felles kjøremiljø stiller det større krav til kvalitet og kontroll på leveranser og idriftsettelse av disse. All mottatt programvare skal derfor gjennom et, på forhånd, definert testopplegg. Avhengig av hvilken type programvare vi mottar, vil det forefinnes egne prosedyrer for mottak. I tillegg vil det finnes spesielle testprosedyrer som vil teste levert funksjonalitet og kvaliteten på dette.

Det skal finnes minst 4 forskjellige programvaremiljøer i Helse Midt-Norge. Disse vil være fordelt med to i hvert av programvare områdene *Utvikling* og *Produksjon.*

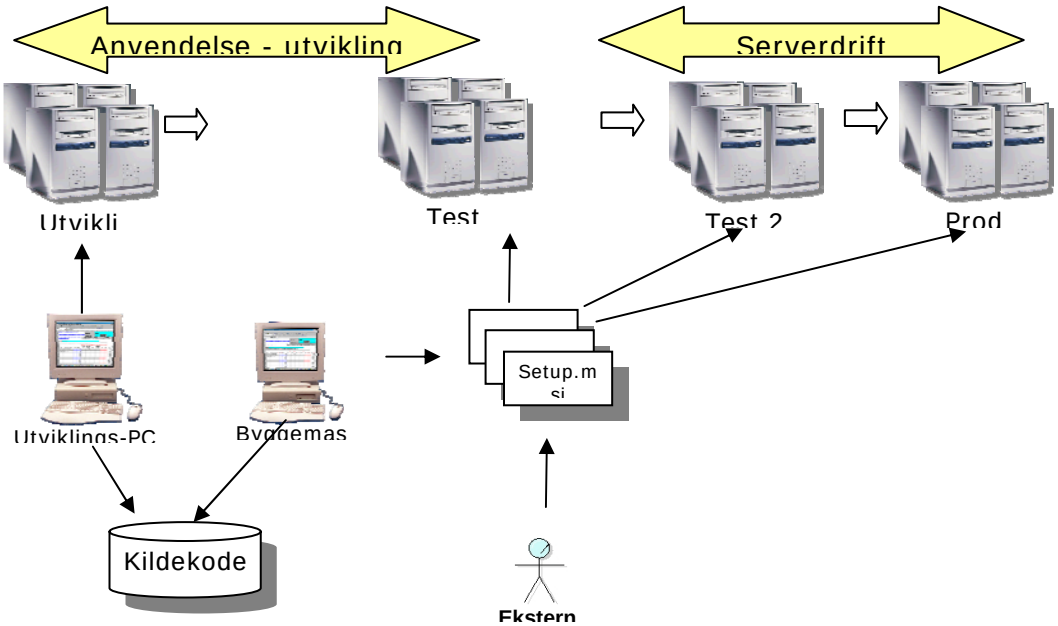
I programvare området *Utvikling* vil det finnes to programvaremiljøer, et utviklingsmiljø og et testmiljø.

Utviklingsmiljøet vil brukes til utvikling av egne løsninger, samt for sjekking av kildekode mottatt av andre leverandører.

Testmiljøet vil brukes til testing av de forannevnte løsningene.

Maskinparken i dette området kan være mindre kompleks og ytelsesmessig dårligere enn i Produksjons området.

I programvare området *Produksjon* vil det også finnes to programvaremiljøer, et testmiljø og et produksjonsmiljø.



Testmiljøet vil brukes til testing av alle løsninger som blir levert for produksjon i Helse Midt-Norge. Testingen vil foregå ved at ansvarlig for installasjonen tester at installasjonsjobbene fungerer som de skal uten manuell inngripen.

Det forutsettes at alle installasjoner leveres som MSI installasjons script. Disse skal være utformet slik at det ikke trengs noen form for redigering/oppretting i etterkant. All dialog/input som kreves skal skje i MSI scriptet. Databaser skal opprettes eller endres vha. databasescript. MSi skriptene skal være konfigurerbare slik at installasjonen kan foregå ”stille”. Dvs. at parametrene som trengs skal kunne angies i en inputfil til msi scriptet. MSI scriptet skal dokumenteres, slik at det til enhver tid er klart hvor scriptet legger ting og hvilke registry nøkler som blir installert/modifisert.

Det skal angies i installasjonsdokumentet hvilket patch nivå applikasjonene krever på operativsystem og andre komponenter som løsningen er avhengig av.

Produktene skal ha avinstallasjonscript som fjerner alle spor etter installasjonen på maskinene.

Tjenesteansvarlig vil i ettertid teste installasjonen funksjonelt.

Produksjonsmiljøet vil brukes til full produksjon av løsningene som er godkjente for installasjon i testmiljøet.

Det skal være forskjellige personer/grupperinger som er ansvarlig for de to hovedområdene. Dvs. at egne driftspersoner, som ikke har utviklet løsningene selv, skal stå for testing og iverksettelse av produksjon i programvareområdet produksjon.

Under hele prosessen skal prosedyrene for utvikling, testing og produksjon følges. Det skal forefinnes logger for hvert av trinnene i prosessen, slik at det er mulig å spore endringer/feil gjennom livssyklusen til et system.

For øvrig henvises det til Implementasjonsguiden for Helse Midt-Norge for en mer detaljert beskrivelse av hvordan leverte løsninger skal integreres med eksisterende systemer.

3 Forretningsmodell/kravspesifisering

3.1 Innledning

Det er startet et arbeid med å innføre en prosessmodell for Helse Midt-Norge. En prosessguide er under utvikling, i første omgang for Hemit, med mål om at den skal bli gjort gjeldende for hele Helse Midt-Norge. Denne vil i første omgang kun omfatte forretningsmodellering og kravhåndtering. På sikt kan prosessmodellen utvides til også å beskrive for eksempel versjons- og konfigurasjonsstyring, testing, endringshåndtering. En prosessmodell for Hemit utarbeides i løpet av Q1 2008.

3.2 Forretningsmodell

Gjennom å ta i bruk standardisert verktøy for utarbeidelse av forretningsmodeller vil man oppnå bedre kommunikasjon med kunder og leverandører. Forretningsmodellene er diagrammer og skisser over arbeidsprosesser i virksomheten og beskriver hva virksomheten gjør og hvordan. Forretningsmodellene skal gjøres tilgjengelig på web og skal ”henge sammen” slik at endringer lett skal kunne spores. Allerede utarbeidete forretningsmodeller skal være lett tilgjengelige for bruk i nye prosjekter. Kravhåndtering

Kravhåndtering er en systematisk fremgangsmåte for å identifisere, dokumentere, organisere og spore kravene i et system. Med sporbarhet menes evnen til å spore ulike prosjektelementer til hverandre, spesielt de som kan relateres til krav.

Hemit’s prosess for kravhåndtering inneholder i hovedsak elementer som interessentbehov, produktegenskaper og detaljerte krav (Use Cases). Prosessen er beskrevet i prosessguiden som inneholder bl.a. aktivitetsbeskrivelser, rollebeskrivelser, maler, retningslinjer og eksempler.

4 Gevinstpotensiale

Det finnes pr. i dag ingen statistikk som er mulig å benytte til en inngående kost/nytte analyse og det vil kreve uforholdsmessig tid og ressurser å utføre en slik analyse. Det er derfor ikke her utarbeidet finansielle gevinstestimater, men det er identifisert 6 ulike gevinstområder.

For hvert område er det gjort en generell vurdering av gevinst potensialer (stort, middels eller lite) i henhold til følgende definisjoner:

Stort gevinst potensial, finansielle gevinster vil sannsynligvis overstige kostnader i perioden

Mindre gevinst potensial, finansielle gevinster vil sannsynligvis tilsvare kostnader i perioden

Lite gevinst potensial, kostnader vil sannsynligvis overstige finansielle gevinster i perioden

Gevinstområder	Beskrivelse	Gevinstpotensialer Kort sikt - 2007	Gevinstpotensialer Lang sikt – 2008- 2010
Utvikling og integrasjon	• Enhetlig strategi for nye tekniske løsninger – SW / HW (Gjenbrukbarhet) • Motiverte og kompetente medarbeidere (Organisasjon) -> Hemit matcher leverandører.	 Reduserte integrasjons-kostnader ved gjenbruk av tjenester og standardiserte metoder.	 Kravspesifikasjon av løsninger. Økt kompetanse på egenutvikling.
Drift	• Serverkonsolidering • Logg/Overvåking Proaktivt kontra reaktivt Enhetlig overvåking -> konsolidering • Ressurser - driftspersonell • Brukeradministrasjon	 Serverkonsolidering. - driftsgevinster (eks. backup, overvåkning) - lisenser (eks. SQLserver) Gevinst øker etter hvert som nye systemer settes i drift.	 Pr. i dag brukes 1-2 årsverk til brukerhjelp i forb. med passord, glemt passord/bytte passord.
Kvalitet og sikkerhet	Kvalitet: • Tilgjengelighet - riktig data til riktig tid • Konsistente data - ikke dobbelregistrering Sikkerhet: • også Mål for IKT-Entreprisen • Sikkerhet - riktig data til riktig bruker	 Bedre konsistens med økt integrasjon med PAS og journal.  Bedret tilgangskontroll ved bruk av ny SSO-tjeneste.	 Dataintegritet. Tilgjengelighet. 
Organisasjon	• ”Single SignOn” SSO • Rolle- og tjenestebasert portal Understøtter arbeidsprosesser En arbeidsflate Potensialer ved å endre arbeidsprosesser	 Brukere av OpPlan og PAS får en pålogging når ny omskrevet versjon settes i drift. Flere systemer forventes å benytte ny SSO-tjeneste.	 Fremhenting av pasient-informasjon fra ulike systemer fra samme arbeidsflate.
Nasjonal samhandling	• Integrasjon • Utveksling av data • Tjenesteutveksling • Felles standarder	 Utarbeide nasjonal arkitekturstandard.	 Nasjonal arkitektur-standard.

5 Veien videre - anbefalte tiltak

5.1 Arkitekturprosess

Arkitekturdokumentet skal være et levende dokument som må vedlikeholdes med jevne mellomrom. Minst en gang i året bør det foretas en revisjon av dokumentet for å holde det à jour i forhold til den tekniske utviklingen. Arkitekturmodellen bør revideres i etterkant av piloten og oppdateres med erfaringer fra denne.

Når det gjelder selve programarkitekturen, så bør det lages en plan for å få flere tjenestebaserte funksjoner gjort tilgjengelig. Dette vil bla. si at flere av de viktigste systemene på sykehusene bør gjøres om i henhold til en tjenestebasert arkitektur. Det er anskaffet en infrastruktur som løsningene kan kjøre i. Denne må videreutvikles med mer fokus på drift og overleverings rutiner.

Det må satses videre på å få utviklet alle fellestjenestene, slik at det blir lettere å implementere nye løsninger basert på arkitekturen. Her er katalogtjenesten særdeles viktig.

Arkitekturarbeidet bør videreføres. Gjerne gjennom en arkitekturgruppe som har faste medlemmer og arbeidsplan. "IT arkitekten" bør ha en rolle i alle prosjekter for å sikre at Helse Midt-Norges arkitekturstrategi etterleves. Dette gjelder både interne og eksterne leveranser.

5.2 Forankring

Forankring er en suksessfaktor. Arkitekturen bør gjøres kjent både internt i Hemit og blant andre aktører som kan bli berørt av denne. De viktigste er sykehusene, det regionale foretaket og leverandørene til foretaket. Internt i Hemit må alle kjenne til grunntrekkene i arkitekturen, slik at de vet hva som kreves av systemene og hva som kreves av kjøremiljøet. De må også kjenne til alle fellestjenestene og deres betydning for systemene.

5.3 Infrastruktur (server og nett)

Etablering av skissert kjøremiljø i versjoner for utvikling, test, test leveranse og drift er gjennomført. Testmiljø kan tilbys eksterne leverandører slik at de kan teste sine systemer mot Helse Midt-Norges fellestjenester. Det må fremover legges vekt på drift og overleveringsrutiner rundt plattformen. Også sikkerhetsrutiner, SSO integrasjon, og portaltjenester bør være i fokus fremover.

5.4 Driftsprosedyrer (serverdrift) – installasjon og endringskontroll

Arkitekturmodellen legger opp til gjenbruk av tjenester og infrastruktur. I en slik sammenheng er kvalitet og kontroll rundt leveranse, utrulling og drifting viktig. I denne sammenhengen bør det defineres rutiner og prosedyrer for egenutvikling, test, mottak av eksterne løsninger og drift av disse.

Følgende områder bør få egne rutiner:

- Utvikling og test av egne løsninger og fellestjenester
- Mottak og test av egne og eksterne leveranser
- Mottak og test av egne og eksterne oppdateringer
- Drift og overvåkning av løsninger i felles kjøremiljø i henhold til ITIL prosessmetodikk.

5.5 Kommunikasjon med Helsebyggs IKT-entreprise (Telenor)

Nytt sykehus er under oppføring i Trondheim i Helsebygg Midt-Norges regi. I den forbindelse er Telenor ansvarlig for utbygging av IKT-infrastruktur. Helse Midt-Norges teknologi og arkitekturstrategi skal samkjøres med Helsebygg-Entreprisen. Ansvarlig for samkjøringen er St. Olavs Hospital HF.

5.6 Kjør en større del av prosjekter/skreddersøm/avdelingssystemer internt

Vedlikehold og kompetanse på arkitekturmodell bør sikres. Det kan gjøres ved å kjøre flere/større prosjekter i eget hus. Som et minimum må utviklere og arkitekter har kjennskap til å være i stand til å videreutvikle fellestjenester etter som man ser behov for slike. Man må også ta stilling til om en portal leveranse skal egenutvikles eller bestilles eksternt. Uansett må dette foregå under egen kontroll.

5.7 Klargjør ansvarsfordeling

Arkitekturmodellen berører i dag flere avdelinger i Hemit. Spesielt videreføring av arkitekturen, som gjelder både Anvendelse og Infrastruktur (serverdrift). Ansvar og forankring bør sikres og avklares.

Ansvar for oppfølging av dokumentasjon og krav til dokumentasjon bør ligge hos IT arkitekten, men må avklares.

5.8 Overordnet informasjonsmodell

En informasjonsmodell skal vise hvordan data logisk hører sammen. Hvordan de oppstår, benyttes og endres. Informasjonsmodellen bør som minimum dokumentere samspillet (inn, ut, metode) mellom de virksomhets- og avdelingskritiske systemer. Elementer som ***eierskap til informasjon*** (ansvar for konsistens, bruk og oppdatering av informasjon), **integrasjon** mellom ulike grupper av systemer og informasjonsutveksling mellom kliniske og pasientadministrative systemer bør berøres.

En overordnet informasjonsmodell bør utarbeides og det bør lages en mal for bruk ved utvikling av nye systemer.

5.9 Sikkerhetsstrategi

Det er etablert en sikkerhetsarkitektur som blir implementert i piloten. En fullstendig sikkerhetsstrategi med fokus på sikkerhetsrutiner, rollemodeller, PKI, sertifikater og kryptering med mer er ikke utarbeidet.

En sikkerhetsstrategi med kartlegging av rollemodeller bør utarbeides. Med forbehold om autorisering, vil dette arbeidet bli videreført gjennom Helsebygg-Entreprisen og med St. Olavs Hospital HF som ansvarlig aktør. Dersom autorisering ikke inngår i det arbeidet, må det tas hånd om i et eget prosjekt.

Vedlegg A: Oppsummert tiltaksliste

Oppsummering av Kap. 5 Vegen videre - anbefalte tiltak

<table>
<tr><td>1</td><td>Arkitekturprosess – arkitekturgruppe/arkitekturforum
Det må defineres en gruppe på 2-4 personer som har et særskilt ansvar for å forvalte arkitekturstrategien og bistå IT-arkitekten i å sørge for at den overholdes og kommuniseres med ulike parter.</td></tr>
<tr><td>2</td><td>Arkitekturprosess - strategi
Arkitekturmodellen bør revideres i etterkant av piloten og oppdateres med erfaringer fra denne.
Arkitekturstrategien må settes opp i en plan for årlig revisjon av strategidokumenter.</td></tr>
<tr><td>3</td><td>Arkitekturprosess - tjenester
Sett i sammenheng med pågående og nye prosjekter bør det settes opp en plan for å få flere tjenestebaserte funksjoner gjort tilgjengelig. Samtidig må det også lages en plan for å få utviklet alle fellestjenestene, spesielt katalogtjenesten.</td></tr>
<tr><td>4</td><td>Forankring
Arkitekturstrategien skal presenteres i avdelingsmøter eller andre relevante fora internt i Hemit, i møter med kundene, i det regionale helseforetaket.
Ansvarlig: IT-arkitekt/IT-sjef Anvendelse</td></tr>
<tr><td>5</td><td>Infrastruktur (server og nett)
Avdeling infrastruktur må, i samarbeid med IT-arkitekten, etablere de nødvendige kjøremiljøene for utvikling, test, test leveranse og produksjon. Testmiljø må tilbys eksterne leverandører for nødvendig testing mot fellestjenestene. Dette kan gjerne skje samtidig med tilrettelegging for drift av nye systemer (for eksempel RoS).</td></tr>
<tr><td>6</td><td>Driftsprosedyrer (serverdrift) - installasjon og endringskontroll
Følgende driftsprosedyrer må lages
<table>
<tr><th>Prosedyre</th><th>Ansvarlig</th></tr>
<tr><td>Utvikling og test av egne løsninger og fellestjenester</td><td>Anvendelse - utviklingsgruppen</td></tr>
<tr><td>Mottak og test av egne og eksterne leveranser</td><td>Anvendelse og Infrastruktur i samarbeid</td></tr>
<tr><td>Mottak og test av egne og eksterne oppdateringer</td><td>Anvendelse og Infrastruktur i samarbeid</td></tr>
<tr><td>Drift og overvåkning av løsninger i felles kjøremiljø</td><td>Infrastruktur – serverdrift/nettos</td></tr>
</table></td></tr>
<tr><td>7</td><td>Kommunikasjon med Helsebyggs IKT-entreprise (Telenor)
Det må etableres kontakter mellom IKT-entreprisen og IT-arkitekten eller arkitekturgruppen.</td></tr>
<tr><td>8</td><td>Kjøre en større del av prosjekter/skreddersøm/avdelingssystemer internt
1. egenutviklingsprosjekt forventes å være Kvalitetsregister for Hjerteinfarkt.</td></tr>
<tr><td>9</td><td>Klargjør ansvarsfordeling mellom IT-arkitekt, arkitekturgruppe, Anvendelse og</td></tr>
</table>

	Infrastruktur. Kan gjøres vha. en prosedyrebeskrivelse.
10	Overordnet informasjonsmodell Rådgiver integrasjon er ansvarlig for at det utarbeides en informasjonsmodell for de eksisterende systemer og en mal til bruk for nye systemer. Arbeidet skal utføres høsten 2004.
11	Sikkerhetsstrategi En fullstendig sikkerhetsstrategi bør utarbeides av sikkerhetsansvarlig, i samarbeid med IT-arkitekten.

Vedlegg B: Ordliste

AD	Active Directory, Microsoft operativsystem komponent
AM	Authorization Manager, Microsoft komponent
ADAM	Active Directory in Application Mode, Microsoft komponent
Applikasjonsarkitektur	Oversikt over systemer, komponenter og tjenester og hvordan de spiller sammen i en virksomhetsarkitektur.
Autentisering	Sjekk på at du er den du utgir deg for å være
Autorisasjon	Å få tilgang til å utføre visse oppgaver
Distribuert N-lagsarkitektur	Lagdelt arkitekturmodell der presentasjon, forretning og datalag kjører atskilt i en distribuert infrastruktur.
EDI	Electronic Data Interchange. Elektronisk utveksling av data mellom IT-systemer.
Fellestjenester	Med fellestjenester mener vi tjenester HELSE MIDT-NORGE tilbyr alle applikasjoner og løsninger. Eks. kan være logg og pålogging.
Forretningsarkitektur	Produkter, tjenester og prosesser som synliggjør forretningsverdier i en virksomhetsarkitektur.
Informasjonsarkitektur	Modell som viser viktigste data og hvilke systemer som eier dem og hvordan de hører sammen i en virksomhetsarkitektur.
Komponent	Byggekloss i en applikasjon eller en løsning. Funksjonalitet som ikke er tilbudt med annen teknologi enn WebServices.
Legacy system	Fagsystem (eksisterende, ”gammelt” system)
LDAP	Lightweight Directory Access Protocol. Standard protokoll for tilgang til katalogtjenester.
Løsning	
N-lagsarkitektur	Lagdelt arkitekturmodell som viser hvordan presentasjon, forretning og datalag skilles.
Programvare arkitektur	Oppbygning av en applikasjon.
SOAP	Simple Object Access Protocol (SOAP) bygger på XML. Denne teknologien gjør det mulig med anrop mellom ulike programmer på en løst sammensatt standardmåte, som igjen gjør det mulig å bygge programmer som distribueres over Internett.
Teknologiarkitektur	Modell over infrastruktur og teknologi for å implementere en virksomhetsarkitektur.
Tjeneste	Funksjonalitet tilgjengelig som en WebService.
Trace	Spor. Sporbarhet (Eng: Traceability) Evne til, ved hjelp av nedtegnet identifikasjon, å spore forløpet, anvendelsen eller lokaliseringen av en gjenstand eller aktivitet, eller lignende gjenstander eller aktiviteter. Her benyttet for å spore forløp i applikasjoner.
UDDI	Universal Description, Discovery and Integration. I en UDDI katalog kan man beskrive og vise hvor tjenester er tilgjengelig.

	Den gjør det enkelt og raskt å finne tilgjengelige tjenester man kan kommunisere med.
Virksomhetsarkitektur	Komplett arkitektur som beskriver alle IT-systemer og hvordan de spiller sammen og hvordan de skal designes for å oppnå dette.
VPN	Virtual Private Network. Sikker oppkobling mot et nettverk (for eksempel, arbeidssted) fra et annet sted (hjemmefra) over internett.
WSDL	Web Services Description Language (WSDL) er en standardmetode for beskrivelsen av ulike standardfunksjoner som er tilgjengelig via visse XML Web services, og hvilke variabler som må overføres for å anrope tjenestene.

EQS HEMIT Helse Midt-Norge IT

Torill Kristiansen

Utform dokument

- Vis dokument
- Innhold
- Nøkkelinfo
- Relatert
- Kommentar
- Høring
- Godkjenning
- Slett revisjon

Under utforming (1)

Hjelp

Under utforming

Id		Tittel	Ver.	Påbegynt	Funksjon	Rolle	Status
1889		IT arkitekturstrategi Helse Midt-Norge	2.0	09.12.2009	Velg funksjon		Under utforming