

1.4 Krav og forventninger

Hvilke krav, forventninger til funksjonalitet, automatisering, sanntid (realtime), drift og oppgraderinger har kunden/brukeren til løsningene.

Kartlegging

Hvis vi forutsetter full oppslutning i organisasjon, eierskap i ledelsen og ingen politiske hindringer, kan en planlegge løsningen integrasjonen skal dekke. Det betyr kartlegging og plan for funksjonelle løsninger, automatiseringsgrad, sanntidsbehov, sikkerhet, type, kapasitet, driftssituasjon, feilhåndtering, mm.

Som nevnt i kapittel Kompleksitet (kap 1.2), kan det være utfordrende å definere hvilke funksjoner som skal integreres, med tilhørende informasjonsinnhold. Årsaken er at vi kort og godt ikke har et detaljert bilde av dette i forkant, kun at systemene skal samspille og at vi skal unngå dobbeltregistreringer. System-integrasjoner vil ofte være en prosess hvor behov, muligheter, utfordringer og løsninger fremkommer underveis i prosjektet. Dermed blir det nesten håpløst å estimere og ikke minste kontraktfeste løsningen med anbud og fastpris. Faren for å måtte gange budsjetter med PI eller SKO-NR blir ofte større jo mer du låser kontrakter til en kravspesifikasjon i et tidlig stadium. Dette er en stor utfordring i virksomheter som må forholde seg til offentlige anbud, hvor tillit, kultur og kompetanse ikke er lett å kvantifisere.



Ut fra dette blir kartleggingen langt viktigere gjennom å undersøke hvilke integrasjonsgrensesnitt, teknologi og arkitektur som ligger som fundament for de enkelte system som skal integreres, ikke minst for kjernesystem som vi må forutsette ikke kan byttes ut eller endres på de nærmeste årene. I tillegg kartlegge kompetansen og kapasiteten hos aktørene som er berørt, være seg brukere, prosjektdeltakere og systemleverandører. Flexibilitet og leveringstid på utvikling/tilpasning vil normalt være avgjørende, med mindre vi snakker om så enkle integrasjoner at ferdige standard-grensesnitt kan eller må benyttes uten endringer.

Funksjonalitet

Skal en kunne lage en komplett funksjonsbeskrivelse på hvilke data som skal utveksles i form av opprettelse, oppdatering og sletting, må en kjenne funksjonaliteten i begge systemene, en kompetanse du sjeldent finner oss en aktør. Skal dette utarbeides av en integrasjonspartner, kjenner denne aktøren kanskje ikke til noen av systemene, kun hvordan en synkronisering rent teknisk kan løses (ref integrer bil og campingvogn). Kartleggingen av funksjonaliteten må da gjerne skje i ett tett samarbeid mellom systemeier/brukere og systemleverandør fra alle/begge aktører, slik at en for hver funksjon finner hvilken informasjon ligger tilgjengelig, i hvilket format og hva er så hensiktsmessig å forsøke å utveksle. Det kan være en fordel å bygge løsningen sten for sten, da en gjenbraker erfaringene fra første delen av løsningen som fungerer. Hvis en for eksempel ønsker 2-veis synkronisering av produkt-register imellom 2 system, blir en av utfordringene hjelpe-tabeller og styringsparametere med tilhørende logikk. Dette gjør til slutt at kravet om 2-veis integrasjon bør fjernes og at produktene vedlikeholdes kun i ett system (ERP). Igjen eksempel på hvor dårlig rigide kontrakter kan fungere.

Automatisering

Hvilke prosesser og funksjoner en ønsker å automatisere, samt kost/nytteverdien av disse må evalueres. Automatisering er egnet på store transaksjonsvolum, med faste og helst standardiserte format. [EDI](#) og [EHF](#) format har det vært jobbet med siden tidlig på 70-tallet, hvor gjerne kun faktura-transaksjoner har fungert bra bla ut fra offentlige og internasjonale krav og standarder, mens spesielt logistikkrelaterte transaksjoner har vært krevende å standardisere.

Selv om manglende automatisering ved hjelp av integrasjoner medfører dobbeltarbeid, må en ut fra kost/nytte vurdere verdien av en elektronisk løsning, hvor manuelle integrasjoner være like effektivt. God arbeidsrutiner for dobbeltarbeid kan da med lite volum fungere godt nok.

Sanntidsløsninger (realtime).

- *Hvor ofte data skal synkroniseres?*
- *Er et umiddelbart svar eller bekreftelse på spørringer et sentralt krav for integrasjonen?*
- *Skal lagring eller oppdateringer skje direkte, også kalt «synkron integrasjon»?*

Et oppslag på lagerverdi kan i noen sammenhenger fungere kun ved at lagersaldoer (disponibelt) overføres kun en gang i døgnet til bruk i for eksempel Netthandel, mens i noen sammenhenger er realtime lageroppslag et absolutt krav. Hvis ikke systemet en skal integrere seg mot, gjerne via internett, har en moderne systemarkitektur med tilhørende API (Application Prover Interface) vil mulighetene får å innfri kravet være tilnærmet umulig. Kvaliteten på feilhåndtering vil også henge tett sammen med type integrasjonsgrensesnitt. Type integrasjoner og API omhandles i kap 1.6 og 1.7.

Sikkerhet

Kravet til sikkerhet i en integrasjon er på linje med øvrige system-løsning og applikasjoner styrt ut fra risiko, datainnhold, brukervennlighet og budsjett. Jo mer låser, krypteringer og koder en legger på noe som skal åpnes for trafikk, jo mer kan dette komme på bekostning av drift, hastighet og brukervennlighet.

Skal trafikken gå via internett eller på annen måte eksternt, er kravet til sikkerhet langt større.

Tekniske løsninger som 2-faktor, web-server, DMZ, Proxy, https og VPN omhandles i kap 1.8.

Type

Tekniske rammebetingelser er som regel det mest avgjørende for valg av type integrasjon, om det skal skje manuelt, fil-basert, meldingsbasert (MSMQ) eller sanntids-API (Soap og Restful). Mer om dette i kap 1.7.

Kapasitet

www.Elhub.no er en nøytral datahub som håndterer alle måledata og markedsprosesser i det norske kraftmarkedet. Utrvekslingen er meldingsbasert og med ett slikt formidabelt volum er planlegging av kapasitet avgjørende, rett og slett for å kunne greie å motta/send alle meldingene ila 24 timer. Mer om caset Elhub i [kap x.x](#)



Driftssituasjon og feilhåndtering

Type integrasjon og teknologi er avgjørende for hvordan integrasjonen skal driftes. Fil- og meldingsbaserte løsninger setter gjerne krav til overvåking, epost/sms-melding, meldingskopi, sletting og loggfiler.

Med direkte integrasjon og bruk av realtime API (soap/rest) blir driftsløsning og feilhåndteringen langt enklere og mer driftssikkert. Integrasjonen vil da i hovedsak bli styrt fra eksternt integrert system.

Mer om feilhåndtering i kap 2.9

Oppgraderinger

En av de store utfordringene for systemeier/kunder og systemleverandørene er oppgraderinger, være seg ny versjon av vertssystemet (for eksempel EPR), integrert løsning (Netthandel, CRM, etc) eller selve integrasjonen. Her skal oppgraderingen nærmest skje i fart, med ingen eller minst mulig nedetid. Stopper integrasjonen kan det ha katastrofale følger for virksomheten, som stans i lab rekvisisjoner og prøvesvar, kasse-terminaler i kjøpesenter/butikker, bestillinger i netthandel, osv.

Dårlig planlegging og gjennomføring av oppgraderinger kan ha større konsekvenser for virksomheten, enn at hovedsystem har nedetid. Manglende oversikt over tiltak, omfang og konsekvenser ved oppgraderinger kan medføre en risiko som tilsier at oppgraderinger utsettes eller aldri gjennomføres. Konsekvens av dette blir at systemløsningen stivner og utgår på dato.

På linje med driftssituasjon og feilhåndtering, vil kvaliteten på integrasjonsløsningen være avgjørende for risiko og omfang av aktiviteter ifm en oppgradering. En sentral sak for integrasjonsgrensesnitt og API er at de er det som kalles Bakover-kompatibel, noe som omtales i kap 2.8 *Oppgradere eksisterende integrasjon som er i full produksjon*