

Johannes Brodwall
Trondheimsveien 18D
0560 Oslo

Oslo, 25. september, 2008

Svar på høring av arbeidsgrupperapport om felles IKT-arkitektur

Dette brevet er et svar på høringen for Felles IKT-arkitektur i offentlig sektor. Jeg sender dette svaret som privatperson selv om jeg er i ferd med å tiltre en stilling i en bedrift som leverer konsulenttjenester til offentlig sektor. Dette er fordi min arbeidsgivers rent økonomisk særinteresser i denne saken ikke er sammenfallende med mine interesser som borger, skattebetaler og kunde av offentlig sektor.

Bakgrunnen for mitt svar er min erfaring fra et stort prosjekt (flere hundre årsverk) i privat sektor, der jeg var en av tre arkitekter med overordnet ansvar for prosjektet. I dette prosjektet hadde jeg blant annet teknisk ansvar for felleskomponenter og gjenbruk. I tillegg til denne erfaringen bygger svaret på den erfaringer jeg har innhentet fra deltagere og foredragsholdere på ikke-kommersielle faglige arrangementer jeg har vært med å organisere, spesielt faggruppen Oslo XP meetup, hvor jeg har arrangert månedlige møter i en årsrekke.

For å beskytte min nåværende og fremtidige arbeidsgivers interesser, har jeg anonymisert erfaringene jeg referer til i dette svaret.

Hovedkonklusjon og anbefalinger

Det er mange gode tanker i rapporten, men disse er svekket av at målene ikke lar seg etterprøve. Jeg utelater å kommentere på gode momenter og fokuserer i stedet på de alvorligste manglene.

Rapporten sannsynliggjør ikke at de anbefalte tiltakene vil ha de foreskrevne effektmålene. Videre gjør rapportens form det vanskelig å etterprøve hvor vidt tiltakene har den ønskede effekten. Jeg ønsker å påpeke følgende konkrete mangler med rapporten. Disse manglene blir begrunnet i resten av dokumentet:

1. Alle effektmål må gjøres mulige å etterprøve
2. Effektmålene av tjenesteorientert arkitektur må beskrives (og gjøres etterprøvbare)
3. Rapporten bør dokumentere graden av måloppnåelse fra et bredt sett med organisasjoner som har gjennomført SOA-prosjekter
4. Rapporten undervurderer kostnaden forbundet med å utvikle og vedlikeholde fellestjenester og bør begrense antall fellessystemer som anbefales dramatisk
5. Rapporten undervurderer kostnaden og overvurderer effekten av å *gjenbruke* fellestjenester, spesielt fellestjenester som må tilpasses hvert bruksområde
6. Rapporten bør legge åpen kildekode/fri programvare til grunn som mekanisme for å oppfylle arkitekturprinsippet om åpenhet og som styringsmodell for felleskomponenter

Etterprøvbarehet av effektmål

Overordnet lider rapporten av at den er for lite konkret. Effektmålene er ikke målbare og arkitekturprinsippene kan dermed ikke etterprøves. Kapittel 7.1.2 og 7.1.3 beskriver ikke-økonomiske vurderinger av effektiviteten av de foreslåtte tiltakene. Denne effektiviteten er imidlertid ikke konkretisert til noe som kan etterprøves. Dette øker risikoen for at store leverandører av verktøyer og konsulenttjenester benytter rapporten som dekning for salg av produkter og tjenester som kan vise seg å ikke være formålstjenlig for det offentlige.

Som et konkret tiltak bør alle effektmål i rapporten gjøres målbare, slik at man kan etterprøve effekten. For eksempel kan "interoperabilitet" måles blant annet i kostnaden for en ny forvaltningstjeneste for å utvikle en løsning som henter ut fellesdata fra et sentralt register, "tilgjengelighet" kan måles blant annet i antall henvendelser som mottas av brukere som ikke mestrer å benytte en digital tjeneste og "sikkerhet" ved hvor lang tid et uavhengig sikkerhetsselskap trenger for å hente ut sensitiv informasjon. Jeg anbefaler at man i videre arbeid forespør Tom Gilb om innspill på dette temaet. Han er en internasjonalt anerkjent ekspert på området og er basert i Norge.

Tjenesteorientering

Seksjon 3.2.1 beskriver tjenesteorientering som et arkitekturprinsipp. Rapporten konkretiserer imidlertid ikke dette prinsippet tilstrekkelig. Arkitekturprinsippet er også utelatt fra kapittel 3.3, som beskriver effektmål ved arkitekturprinsippene. Spesielt for en teknologi som er så mye omtalt, så uklart definert og så aggressivt markedsført er det meget viktig at rapporten beskriver ønskede effektmål av tjenesteorientering på en måte som er etterprøvable.

"Tjenesteorientering" benyttes ofte som et salgargument for teknologi og tjenester som ikke er moden eller formålstjenlig. Prosjekter som innfører tjenesteorientering har i mange tilfeller rapportert at de ønskede effektene har uteblitt [1], og vi kan anta at mislykkede SOA prosjekter er underrapportert (som for de fleste andre mislykkede prosjekter). Videre arbeid rundt felles IKT-arkitektur bør innhente bred erfaring fra SOA prosjekter og sammenligne prosjektenes resultat med opprinnelig planlagte effektmål.

Seksjon 5.3.7 ("kost/nyttevurdering" av "felles metadata") gjør en viktig observasjon rundt tjenester, men rapporten tar ikke inn over seg konsekvensen av denne erfaringen: "datadefinisjoner i konkrete anvendelser [kan ikke] endres uten å informere andre brukere av informasjonen om endringer i innhold." Denne observasjonen gjelder alle tjenester som er tilbudt til andre organisasjoner. En levert tjeneste medfører permanent økt vedlikeholdskostnad da versjoner av tjenesten som er i bruk ikke kan endres uten å skade alle brukerne av tjenesten. For tjenester som er eksponert på Internett er det umulig å finne og informere alle brukerne. Derfor bør rapporten anbefale at kun tjenester som har en klart definert systemmessig bruker og et klart definert formål skal eksponeres som tjenester.

I strid med observasjonen nevnt over opprettholder rapporten St. meld. nr 17 (2006-2007) seksjon 7.4: "Arkitekturen skal være lagdelt og vil minimum bestå av eit presentasjonslag, ein [sic] felleskomponentlag og eit virksomhetslag". Ved å eksponere virksomhetslaget som tjenester vil man øke utvikling- og vedlikeholdskostnaden for virksomhetens systemer.

Jeg antar at formålet ved tjenesteorientering er å redusere kostnaden ved å benytte offentlige systemer fra andre systemer, det være seg offentlige eller private. Det vil si at man ønsker at Tilgjengeligheten (3.2.3) ikke kun begrenses til menneskelige brukere. Effektmålene ved tjenesteorientering kan etter min erfaring best oppnås ved å behandle krav fra systemmessige brukere på tilsvarende måte som krav fra menneskelige

brukere av systemet, og ved å standardisere formatet på den informasjonen som utveksles der dette er formålstjenelig.

Prinsippet om tjenesteorientering bør også konkretiseres. Det er viktig at det framkommer hvor vidt "tjenesteorientering" skal tolkes som et bestemt sett med teknologier.

Fellestjenester og -komponenter

Etter min erfaring fra prosjekter innen privat sektor er gjenbruk av felles driftede tjenester alltid kostnadøkende for enkeltprosjekter. Ofte er denne kostnadsøkningen dramatisk. Konkret tilsier min erfaring at for eksempel tabell 7.3 er farlig feil. Tabellen mangler en rad for "Ikke bruk av skjemamotor". Min erfaring tilsier at kolonnen "kostnad per skjemamotor" vil for systemer som ikke bruker skjemamotor være lavere enn kostnaden ved en felles anskaffelse.

Dette er viktig å merke seg: Dersom man taper penger for hvert sted man gjenbruker en komponent, er ikke lenger gjenbruk på stor skala lønnsomt.

Gjenbruk av tjenester må forsvarers fra en virksomhetsmessig (stor) gevinst, da det er en prosjektmessig (stor) kostnad. Rapporten forventer en prosjektmessig besparing ved gjenbruk av felleskomponenter, noe min erfaring tilsier at er meget urealistisk.

Unntaket fra denne regelen er tjenester som er enkle og som har et meget veldefinert ansvarsområde der det ikke forutsettes noen tjenestemessig tilpasning. Av forslagene i kapittel 5.3 i rapporten anser jeg det som sannsynlig at kun "Autentisering (eID) og autorisering" og "Innkrevning av mindre avgifter og gebyrer" har et potensiale til kostnadmessig besparing for prosjekter som tar disse i bruk. "Offentlig elektronisk postjournal" og "Felles registerdataforvaltning" vil kunne ha forretningsmessige gevinster som kan forsvare kostnaden, slik rapporten beskriver. "Samtrafikknavn", "Skjemamotor", "Meldingsboks" og "Rammeverk for eDialog" vil, dersom de utvikles som felles *tjenester* sannsynligvis innebære en merkostnad for prosjektene som tar disse i bruk. Jeg er føler meg ikke kompetent til å uttale meg om effekten av de øvrige felleskomponentene. (Litt enkelt kan man si at "*bruk* av en tjeneste kan være lønnsomt, mens *gjenbruk* sjelden er lønnsomt")

Seksjon 5.3.3 i rapporten understøtter denne vurderingen: "Skjemamotoren [i dagens Altinn] er svært tung i bruk samtidig som den er lite fleksible [sic] og ikke støtter utforming av kompliserte skjemastrukturer med store og kompliserte kontrollsett". Min erfaring er at dette er et veldig vanlig problem med denne type fellesløsninger. Rapporten sannsynliggjør ikke at et fremtidig prosjekt for å utvikle en ny felles skjemamotor ikke vil ha samme resultat: En dyr løsning som øker kostnadene for de som benytter den. Rapporten anbefaler altså en løsning som man basert på rapportens eget erfaringsgrunnlag ikke kan forvente at skal være formålstjenelig.

Styringsmodeller

Kostnadene med å utvikle og vedlikeholde gjenbrukbare systemer er alltid høyere enn å utvikle systemer for enbruk [2]. Min erfaring tilsier at gjenbruk av installerte systemer er svært dyrt, mens gjenbruk av enkeltkomponenter ved at disse prinsippmessig kopieres til alle brukere er mye billigere. Noe dokumentasjon refererer til dette som "komponentgjenbruk", men mitt inntrykk er at rapporten benytter begrepet "komponent" på

en annen måte. Jeg vil derfor referere til gjenbruk ved å kopiere en versjon av en komponent til hvert enkeltsystem som skal benytte denne som "biblioteksgjenbruk".

Biblioteksgjenbruk er utsatt for samme risiko som tjenestegjenbruk: Biblioteker av dårlig kvalitet eller som ikke løser de aktuelle problemstillingene kan medføre økte kostnader i stedet for reduserte kostnader. I min erfaring er dette spesielt aktuelt dersom det stilles et krav om at løsningen skal gjenbruk et spesifikt bibliotek eller tjeneste. Da blir ikke leverandøren av denne tjenesten eller biblioteket tvunget til å faktisk levere et formålstjenlig produkt. Min erfaring etter å ha analysert resultatet av sentralstyrt utvikling for gjenbruk er at besparingseffekten uteblir.

Min erfaring er med styringsmodellene beskrevet som 1 og 4 i rapporten. Erfaringen med styringsmodell 4 var at gruppen som mottok et eget budsjett for å utvikle fellestjenester ikke maktet å prioritere leveransene på en måte som tjente de forretningsrettede systemene. Samtidig ble prosjektene påført en betydelig kostnad i form av koordinering med felleskomponentgruppen. Ved modell 1 ble noe potensial for besparelse ikke utnyttet, men prosjektene var i stand til å agere selvstendig. I perioden med denne styringsmodellen ble det levert mindre fellesfunksjonalitet, men denne funksjonaliteten hadde mye større kvalitet og besvarte de reelle behovene mye bedre. Modellen som ble brukt for deling tilsvarte åpen kildekode, der kildekoden kun ble tilgjengeliggjort internt i bedriften.

Jeg vil på det sterkeste anbefale at dersom man benytter en styringsmodell som beskrevet i modell 2 eller 3 i seksjon 6.3 så pålegger man kun *at* fellesløsninger benyttes, ikke *hvilke* løsninger som velges. Modell 4 og 5 vil gavne det private næringsliv som leverer programvare og konsulenttjenester, men få andre. Fellestjenester utviklet sentralt har en høy kostnad og lav måloppnåelse.

Åpen kildekode

De mest vellykkede modellene for gjenbruk handler om åpen kildekode eller fri programvare. Derfor er det beklagelsesverdig at rapporten ikke nevner åpen kildekode ut over ett avsnitt i seksjon 6.6. Nasjonalt kompetansesenter for fri programvare ("Friprog") har gjort mye godt arbeid rundt styring- og finansieringsmodeller av felles programvare, og rapportens kvalitet svekkes kraftig av at dette arbeidet ikke benyttes. Åpen kildekode eller fri programvare er en mekanisme for utvikling av felles programvare som har vist seg langt mer effektiv på å oppnå gevinster ved gjenbruk, uavhengig av hvilken styringsmodell som benyttes. For offentlig sektor har også åpen kildekode et stort potensiale for å oppnå åpenhet i form av transparente løsninger.

Rapporten nevner selv viktigheten ved at "tjenestens logikk og datakilder skal være kjent" (seksjon 3.2.5). Den naturlige konsekvensen av dette ville være å kreve at programvare som utvikles for skattebetalerens regning må være fritt tilgjengelig for publikum, med mindre spesielle hensyn til personvern hindrer dette. I praksis betyr dette åpen kildekode eller fri programvare.

Avslutning

Som borger, skattebetaler og kunde av offentlig sektor ser jeg at offentlig sektor kan klare å levere bedre og billigere IKT-tjenester. Dessverre ser jeg også en risiko for at anbefalingene i rapporten kan tolkes på en slik måte at effekten blir dårligere og dyrere tjenester. Jeg applauderer tiltaket og arbeidsgruppens betydelige innsats i å konkretisere

prinsippet om felles IKT-tjenester i offentlig sektor, og håper at mine kommentarer kan være nyttige for å oppnå målene bak dette arbeidet.

Dersom departementet ser en nytte i dette, stiller jeg meg gjerne tilgjengelig for ytterligere kommentar.

Referanser:

- [1] "- Bli skeptisk straks du hører SOA," av Helge Skrivervik, Digi.no 23. mai, 2008 [<http://www.digi.no/php/art.php?id=530381>]
- [2] "Hidden Costs Of Code Reuse", av Alan Radding, InformationWeek November 9, 1998 [<http://www.informationweek.com/708/08iuhid.htm>]

Om avsenderen:

Johannes Brodwall har jobbet i IT-bransjen som konsulent og arkitekt i 10 år. Han er aktiv i fagmiljøer for Java og smidige metoder. Han har blant annet hold foredrag på JavaZone konferansen i 7 år, arrangerer månedlige fagmøter i Oslo XP meetup og er medarrangør av Smidig 2007 og Smidig 2008 konferansene. Han skriver ofte om programvareutvikling på sin blog på <http://brodwall.com/johannes/blog> og er har publisert flere artikler i blant annet java.net. For mer informasjon, se Johannes' side på LinkedIn: <http://www.linkedin.com/ppl/webprofile?id=1045686>