



# Prosessforbedring og teknologistøtte for semistrukturerte prosesser

Hovedoppgave  
ved  
sivilingeniørutdanning i  
informasjons- og kommunikasjonsteknologi

av  
Erik Tessem



Grimstad 28. mai 2001



# **Prosessforbedring og teknologistøtte for semistrukturerte prosesser**

### Sammendrag

Rapporten omhandler utvikling av et adaptiv Workflow Management System (WfMS). Systemet skal brukes i forbedring av en *faktureringsprosess for konsulenter*. I den forbindelse behandles sentrale teorier innen prosessforbedring og adaptiv arbeidsflyt.

Nåværende faktureringsprosess har rom for effektivisering, automatisering og kvalitetsforbedring. Prosessen er manuell, sekvensiell og lite strukturert. Strategi som foreslås benyttet i utviklingen er Total Quality Management (TQM), Computer Supported Cooperative Work (CSCW) og Concurrent Engineering (CE).

Det er utviklet en prototyp som er verktøyet for å gjøre faktureringen mer automatisert, tilpasningsdyktig og pålitelig. Det lå sterke føringer fra oppdragsgiver i valg av teknologi, MS Outlook 2000, og design. Likevel behandles andre aktuelle teknologier og designløsninger.

Databasen som ligger i bagrunnen er Microsoft Data Engine (MSDE). Den er kompatibel med Ms SQL server. MSDE er et kraftig Database Management System (DBMS) i Ms Access 2000. MSDE er valgt for at systemet lett skal kunne integreres med eksisterende faktureringsystemer.

Rammeverket for funksjonalitet er vist i prototyp. Implementering av funksjonalitet er ikke ferdig (Se vedlegg). Prototyp danner likevel grunnlag for diskusjon omkring utviklingsplattformens og prototyps egnethet for adaptiv prosessforbedring.

**Forord**

Oppgaven, *Prosessforbedring og teknologistøtte for semistrukturerte prosesser*, inngår som min avsluttende hovedoppgave (diplomarbeid) i min sivilingeniørutdanning innen Informasjons- og kommunikasjonsteknologi (IKT), ved Høgskolen i Agder (HiA).

Oppdragsgivere har vært Hewlett Packard (HP) og HiA. HP har definert oppgavens case/problemstilling. HiA ønsket å sette problemstillingen inn i et teoretisk perspektiv med prosessforbedring og adaptiv arbeidsflyt som sentrale tema.

Prosjektet er utført våsemesteret 2001, og varigheten har vært 20 uker. Skolens veileder har vært Lars Line. Kontaktpersoner i HP har vært Joachim Stray og Elisabeth Enoksen. Lars Line har vært teknisk og administrativ veileder, mens Joachim Stray og Elisabeth Enoksen har stått til disposisjon i forbindelse med spørsmål. En takk rettes til de tre for oppfølging gjennom oppgaven. Lars Line rettes en ekstra takk for god veiledning gjennom hele perioden.

Under prosjektet har jeg måttet sette meg inn i prosessforbedringsteori og ny teknologi, Visual Basic og et nytt DBMS, som begge var relativt ukjente teknologier for meg. Prosjektet har vært lærerikt faglig, teknisk og menneskelig.

Erik Tessem

Grimstad 28.05.2001

**Innholdsfortegnelse**

<b>SAMMENDRAG</b>	<b>III</b>
<b>FORORD</b>	<b>IV</b>
<b>1 INNLEDNING</b>	<b>8</b>
<b>2 OPPGAVEBESKRIVELSE</b>	<b>10</b>
2.1 Problembeskrivelse	10
2.2 Mål	10
2.3 Avgrensninger	11
<b>3 TEORIGRUNNLAG</b>	<b>12</b>
3.1 Behovsanalyse	12
3.2 Arbeidsflytsystem og CSCW	13
3.3 Gruppevareteknologi	13
3.4 Tilpasningsdyktig gruppevareteknologi	14
3.5 Informasjonssystem og Virksomhetsprosesser	14
3.6 Virksomhetsutvikling	17
3.7 IKT, IS og Virksomhetsutvikling	17
3.8 Venkatramans rammeverk	18
3.8.1 Det evolusjonære nivå	19
3.8.2 Det revolusjonære nivå	20
3.9 Concurrent Engineering	22
3.10 Integrated Product Development	23
3.10.1 Fordeler med CE og IPD	23
3.11 Total Quality Management(Total KvalitetsLedelse)	23
3.11.1 Konflikter mellom samarbeidsmodellen og TQM	25
3.11.2 Arbeidsflyt og arbeidsflytsystem	26
3.12 Adaptiv arbeidsflyt	29
3.12.1 Klassifikasjon av adaptiv arbeidsflyt	30
3.12.2 Suksessfaktorer for utvikling av et adaptivt arbeidsflytsystem	31
3.12.3 Arbeidsflytteknologier versus manuelle arbeidsprosesser	32
3.12.4 Adaptiv arbeidsflyt i perspektiv til en faktureringsprosess	32
<b>4 PROSESSFORBEDRING</b>	<b>34</b>

---

<b>4.1</b>	<b>Nåværende Rutine for fakturering i HP</b>	<b>34</b>
<b>4.2</b>	<b>Evaluering av nåværende faktureringsløsning</b>	<b>35</b>
<b>4.3</b>	<b>Strategi for prosessforbedring</b>	<b>35</b>
<b>4.4</b>	<b>Forslag til prosessforbedring med arbeidsflytstøtte</b>	<b>37</b>
<b>5</b>	<b>TEKNOLOGI</b>	<b>40</b>
<b>5.1</b>	<b>Microsoft Office 2000</b>	<b>40</b>
<b>5.2</b>	<b>Microsoft Office 2000 Developer</b>	<b>40</b>
<b>5.3</b>	<b>Microsoft Outlook 2000</b>	<b>41</b>
<b>5.4</b>	<b>Utviklingsmetoder i Microsoft Outlook</b>	<b>41</b>
5.4.1	Outlook Forms	41
5.4.2	Office Document Forms	41
5.4.3	Visual Basic for Application (VBA)	41
<b>5.5</b>	<b>Microsoft Office XP</b>	<b>42</b>
5.5.1	SmartTags	43
5.5.2	Forståelige feilmeldinger	43
5.5.3	Task Pane	43
5.5.4	Tekst til tale	44
5.5.5	Tale til tekst	44
5.5.6	Ms Office 2000 vs. Ms Office XP	45
5.5.7	Valg av utviklingsmetode i Outlook	45
<b>5.6</b>	<b>Database</b>	<b>47</b>
<b>5.7</b>	<b>Microsoft Access</b>	<b>47</b>
5.7.1	Jet Data Engine (JDE)	47
5.7.2	Microsoft Data Engine (MSDE)	47
5.7.3	Jet Data Engine (JDE) versus Microsoft Data Engine (MSDE)	47
<b>5.8</b>	<b>Microsoft Exchange 2000</b>	<b>48</b>
5.8.1	Web Storage System	48
<b>5.9</b>	<b>Ms Data Engine versus Ms Exchange 2000</b>	<b>48</b>
<b>6</b>	<b>DATABASEMODELL OG DESIGN AV DATABASE</b>	<b>50</b>
<b>6.1</b>	<b>Valg av databasemodell</b>	<b>50</b>
6.1.1	Oppsummering databasemodeller	51
<b>6.2</b>	<b>Design av databasemodell</b>	<b>52</b>
6.2.1	Prosjekt og Faktureringsrutiner	52
6.2.2	Prosjekt-Kundemodell	54
6.2.3	Mål	55
<b>6.3</b>	<b>Endelig databasemodell</b>	<b>56</b>

<b>6.4</b>	<b>Forutsetninger i forbindelse med design</b>	<b>57</b>
<b>6.5</b>	<b>Databasen i et systemperspektiv</b>	<b>58</b>
6.5.1	Adaptivitet i sanntid	59
6.5.2	Adaptivitet på designnivå	59
<b>7</b>	<b>INNFØRING, PROTOTYP OG PLATTFORMENS EGNETHET FOR ADAPTIV PROSESSFORBEDRING</b>	<b>63</b>
<b>7.1</b>	<b>Prototypens egnethet for adaptiv prosessforbedring</b>	<b>63</b>
<b>7.2</b>	<b>Plattformens egnethet for adaptiv prosessforbedring</b>	<b>66</b>
<b>8</b>	<b>HPFAKTURA – EVALUERING OG DISKUSJON</b>	<b>67</b>
<b>8.1</b>	<b>Feedback</b>	<b>67</b>
<b>8.2</b>	<b>Beslutningsstøtte</b>	<b>67</b>
<b>8.3</b>	<b>Identifikasjonsparametere</b>	<b>69</b>
<b>8.4</b>	<b>Fraværsmodul</b>	<b>71</b>
<b>9</b>	<b>KONKLUSJON</b>	<b>72</b>
<b>10</b>	<b>KILDER</b>	<b>73</b>
	<b>VEDLEGG</b>	<b>75</b>
	Implementert funksjonalitet	75
	Veiledning installasjon	75
	Kildekode	77

### 1 Innledning

Informasjon og kommunikasjonsteknologi (IKT) benyttes i større og større grad for å understøtte eller automatisere arbeidsoppgaver i store og små organisasjoner. Arbeidsflytsystemer er en spesiell kategori av IKT systemer som karakteriseres ved automatisering av arbeidsflyt i prosesser som involverer flere medarbeidere.

Innføring av et arbeidsflytsystem danner grunnlag for nye muligheter og utfordringer. Automatisering brukes til å eliminere manuelle arbeidsprosesser, det vil si eliminere behovet for mellommenn i en prosess.

Det påvirker samhandlingen i en prosess. Automatiseringen danner grunnlag for endring av aktivitetssekvensen for en prosess til mer parallell utførelse av aktivitetene (Concurrent engineering). På den måten tillater et arbeidsflytsystem kostnadseffektive intra- og interorganisatoriske koblinger av aktiviteter.

Utfordringene ligger i innføringsproblematikk og akseptanse av systemet. Når forandringer blir gjort på godt innarbeidede arbeidsprosesser vil det føre til motstand fra brukerne.

Egenskapene ved et system for automatisering av en prosessflyt kan ses i sammenheng med *Computer Supported Cooperative Work* (CSCW). CSCW er et virkemiddel for å oppnå parallellitet i arbeid (Concurrent engineering). Sentrale egenskaper ved CSCW er forståelsen av grupper. Det innebærer forståelse og støtte for samarbeid, samhandling, kommunikasjon og koordinering mellom individer og mellom og innad i grupper og prosesser. Gruppevare er systemer som faller innenfor CSCW systemer.

Oppgaven, *Prosessforbedring og teknologistøtte for semistrukturerte prosesser*, tar for seg faktureringsprosess for konsulenter som case.

Rapportens forskjellige deler er:

- Generell beskrivelse av problemstilling (Kap. 2)
- Sentrale teorier innen prosessforbedring og arbeidsflyt (Kap. 3)
- Prosessforbedring i HP Services' faktureringsrutine (Kap. 4)
- Beskrivelse av teknologi (Kap. 5)
- Databaseløsning (Kap. 6)
- Innføring, prototyp og plattformens egnethet for adaptiv prosessforbedring (Kap. 7)
- Evaluering/Diskusjon av prototyp (Kap. 8)
- Konklusjon (Kap. 9)

Delkapitler der det er nødvendig med en diskusjon drøftes fortløpende. Det danner grunnlag for den endelige diskusjonen rundt HPFaktura mot slutten av rapporten.



### **Bakgrunn**

Mange bedrifter har en hverdag som er dynamisk. Hverdagens dynamiske innhold fører ofte til vedtak som endrer forutsetninger, ressurstilgang, organisasjon og partnerskap. Det medfører at de organisasjonelle prosessene blir usikre, og av den grunn må utvikles. En arbeidsflyt som ikke fanger opp organisasjonelle endringer vil skape problemer. Løsningen kan være adaptiv arbeidsflyt. Det er viktig å følge "føre var prinsippet" for å møte slike organisasjonelle endringer.

Endrede forutsetninger vil oppstå i den dynamiske hverdagen til de store kunnskapsbedriftene, og spesielt teknologibedriftene. Ønsket om økt effektivitet, fleksibilitet og tilpasningsdyktighet i organisasjoner blir overordnet mål ved innføring av et Workflow Management System (WfMS).

## **2 Oppgavebeskrivelse**

### **Prosessforbedring og teknologistøtte for semistrukturerte prosesser**

Hverdagen til moderne kunnskapsarbeidere er dynamisk og usikker. For at arbeidsflytsystemer skal være en effektiv støtte for forretningsprosesser i dette miljøet, må de være i stand til å tilpasse seg når avvik fra "ideelle" prosesser oppstår. Eksempler på avvik kan være vedtak som endrer forutsetninger, endringer i ressurstilgang eller endringer i organisasjon og partnerskap. Dersom arbeidsflytsystemene ikke kan fange opp og endre seg i takt med de ønskede prosesser vil de sannsynligvis gjøre mer skade enn nytte.

Oppgaven utføres i samarbeid med HP Services og HiA. Avdelingen er nyopprettet og er i ferd med å etablere rutiner og prosesstøtte. Avdelingen er i sterk vekst, rutinene må derfor være skalerbare. Som case velges *faktureringsprosess for konsulenter*.

- Gi en oversikt over ulike teorier for prosessforbedring.
- Eksisterende prosess skal kartlegges og settes i sammenheng. Strategi for prosessforbedring skal drøftes.
- Adaptiv arbeidsflyt er et relativt nytt perspektiv innen prosesstøtte. Belys dette perspektivet og drøft relevans til case.
- Foreslå prosessforbedring med arbeidsflytstøtte.
- Lag en prototype for støtte av prosessen. Etter ønske fra oppdragsgiver utvikles prototype i Outlook 2000.
- Innføring av systemet samt prototyp og plattformens egnethet for adaptiv prosessforbedring skal drøftes.

### **2.1 Problembeskrivelse**

Hos HP eksisterer ingen enhetlig struktur for hvordan prosjekter og oppdrag skal faktureres. Jobben er i stor grad manuell og prosessen inneholder mange ledd. Det er ønskelig at konsulentene skal få mer ansvar i faktureringsprosessen. I tillegg har HP behov for en løsning som gir færre rom for feil, samler historisk informasjon og muliggjør beslutningsstøtte.

Løsningen skal fungere som et verktøy for å danne faktureringsgrunnlag. I første omgang skal den ikke integreres med eksisterende systemer, men kun produsere dokumenter som manuelt brukes for å registrere faktura i disse systemene. Videre er det ønskelig å se på muligheter og begrensninger ved en senere integrering med eksisterende faktureringsystem.

### **2.2 Mål**

Prosjektet skal søke å forbedre eksisterende faktureringsprosess for konsulentene ved bruk av CSCW. Det nye systemet skal bidra til å danne en arbeidsflyt som kan tilpasses avvik fra den ideelle prosess, et adaptivt system. Systemet for timeregistrering skal være enkelt og

intuitivt, for å øke akseptansen hos konsulentene. Løsningssystemets utviklingsnavn er HPFaktura.

Arbeidsflyten er et strategisk valg i forbindelse med ledelsens ønske om at konsulentene skal få mer ansvar i faktureringsprosessen. I tillegg har HP behov for en løsning som gir færre rom for feil, samler historisk informasjon og muliggjør beslutningsstøtte.

### **2.3 Avgrensninger**

Det skal ikke utvikles et komplett faktureringsystem. Det er i samråd med veileder og HP Services bestemt at utviklingen i første omgang skal resultere i en enkel prototyp. Den skal danne grunnlag for vurdering av teknologiens egnethet. Spesielt vil det bli lagt vekt på hvilke muligheter og begrensninger teknologien har. Det ligger sterke føringer fra HP Services i prototyps grafical user interface (GUI) og egenskaper. Det har derfor ikke vært mulig for meg å legge inn de løsningene jeg ville foretrukket å benytte i systemet. Løsningene og ideene jeg har tilegnet meg gjennom arbeid med teknologi- og teorigrunnlag for oppgaven vil presenteres i evalueringen av prototyp.

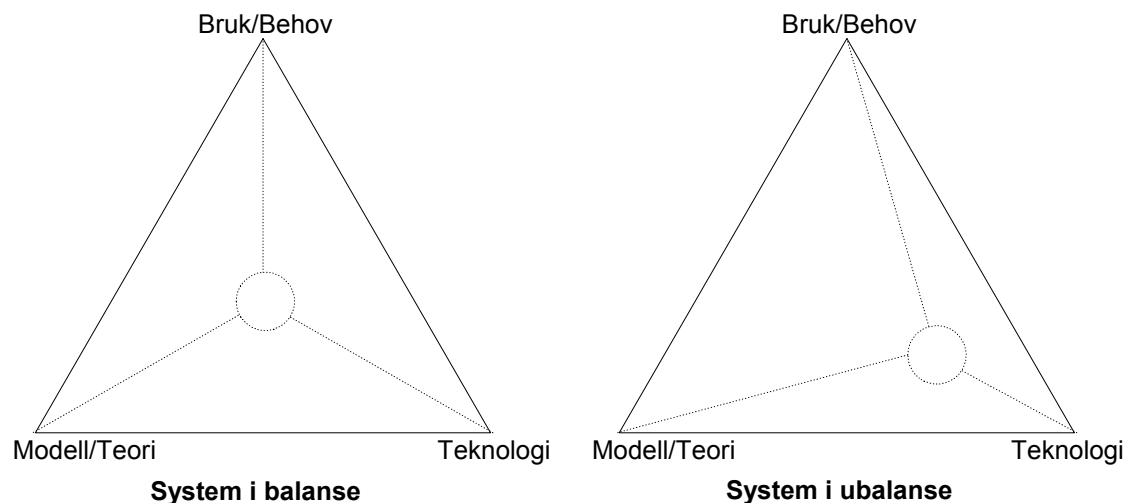
### 3 Teorigrunnlag

Først gis en kort innføring i behovsanalyse. Videre presenteres grunnleggende teorier innenfor prosessforbedring, arbeidsflyt og adaptivitet. Det er ikke gitt at teoriene har direkte relevans til case, *faktureringsprosess for konsulenter*. De skal danne grunnlag for å vurdere og foreslå tiltak til endring av dagens arbeidsflyt, slik at ønsket prosessforbedring og adaptivitet oppnås.

#### 3.1 Behovsanalyse

Behovsanalyse er et virkemiddel for bruk i strategiske valg innenfor IKT teknologi. Når IKT system skal designes og implementeres må det legges ned gode analyser i forkant. Analysen må finne et fornuftig forhold mellom *bruk/behov*, *modell/teori* og *teknologi*. De må være i balanse slik at systemet er i likevekt. Likevekt er første steg på veien til et system som oppfyller forventningene.

Velges en modell/teori eller teknologi som er feildimensjonert i forhold til hverandre og bruk/behov, vil systemet komme i ubalanse. Det gir aldri et godt resultat. Er systemet i ubalanse øker risikoen for at systemet ikke innfrir forventningene. Det medfører redusert aksept til systemet. I et systemperspektiv er et system som ikke tilfredsstillende forventningene mislykket. Figur 1 presenterer et system i balanse og et i ubalanse.



**Figur 1: Bruk/Behov, Modell/Teori og Teknologi i balanse og ubalanse**

Til venstre i Figur 1 er systemet i balanse. Det er harmoni mellom bruk/behov, modell/teori og teknologi. Til høyre er det ingen likevekt mellom faktorene. Resultatet blir et system i ubalanse. Det sees godt ved at fokuset (dvs. sirkelen) flyttes fra tyngdepunktet i trekanten og mot den overfokuserede delen av systemet.

### **3.2 Arbeidsflytsystem og CSCW**

Vi er inne i en tid hvor det i større og større grad blir tatt i bruk IKT systemer for å få utført arbeidet i både store og små organisasjoner. Et arbeidsflytsystem er et IKT system som hjelper til med å få gjort arbeidet raskt, enkelt og effektivt. Et slikt system benyttes ikke av en person, men flere personer i et fellesskap.

Et arbeidsflytsystem er et system som håndterer flyten av arbeidsoppgaver i en organisasjon. Flyten kan være oppgaver som er knyttet sammen for å lage et produkt, en tjeneste eller deler av et produkt eller en tjeneste. Hvilket perspektiv som benyttes tar utgangspunkt i hvilket forhold en har til arbeidsoppgavene.

I slike fellesskap fokuseres det av forskerne på fagfeltet CSCW. Begrepet CSCW ble første gang benyttet av Irene Greif (MIT) og Paul Cashman (Digital) i 1984 på et tverrfaglig workshop [4].

Frem til i dag har begrepet CSCW blitt betydelig utvidet og favner nå flere fagfelt. Tidligere var det fokus på at det var enkeltpersoner som skulle ta i bruk systemene. Det har nå endret seg, slik at systemene designes og utvikles for å gi både en enkeltperson og en gruppe den nødvendige funksjonalitet. Det har ført til at begrepet CSCW nå omfatter begrepet gruppevare.

### **3.3 Gruppevareteknologi**

Teknologi for såkalt mellommenneskelig kommunikasjon, spesielt innenfor etablerte grupper, går under betegnelsen *gruppevare*. Gruppevare brukes til å støtte grupper av individer som er engasjert i løsning av en felles oppgave. Teknologien representerer et grensesnitt til et felles brukermiljø.

Gruppevare er i ferd med å få en stor utbredelse som følge av økt fokusering på prosjektorganisering, gruppe-/teamarbeid, tverrfunksjonelt arbeid og desentralisering. Trenden er at sekvensielle arbeidsoperasjoner erstattes av parallellitet i oppgaveutførelsen.

Gruppevareteknologien gir personer mulighet for å kommunisere med hverandre uavhengig av tid og sted. Dette har vist seg å være et nyttig hjelpemiddel fordi organisasjonene er preget av desentralisering, og at den enkelte ansatt får større frihet med hensyn til arbeids -sted (rom) og -tid.

Arbeidsflytsystemer skal støtte kommunikasjon og koordinering uavhengig av sted (rom) og tid. Teknologien kan som nevnt kategoriseres i tid og rom – fra det som går under betegnelsen møteteknologi (samme tid og sted) til koordineringsteknologi (forskjellig tid/samme sted). En referansemodell for gruppevare er presentert i Figur 2

	Samme tid	Forskjellig tid
Samme sted	<b>Møterom - ansikt til ansikt</b> Fysiske møter Elektroniske møter Gruppe beslutningssystem	<b>Pågående oppgaver</b> Oppslagstavle Databaser Dokumentsystem Konferanser
Forskjellig sted	<b>Avstandssamhandling</b> Interaktive konferanse -Videokonferanse -Telefonkonferanser -Tekstkonferanser Deling av skjerm	<b>Kommunikasjon og koordinering</b> Elektronisk post Arbeidsflyt Elektronisk oppslagstavle

**Figur 2: Gruppevareteknologiens tid/sted matrise**

Matrisen deles opp i fire ruter, hvor hver av rutene representerer forskjellige former for gruppevare. *Samme tid og samme sted* representerer teknologi som primært støtter opp om tradisjonell møtevirksomhet. *Forskjellig tid og samme sted* er teknologi som støtter opp om overføring av kunnskap i tid. Typiske eksempler er felles arkiv og databasesystemer. *Samme tid og forskjellig sted* har behov for konvensjonelle kommunikasjonsverktøy. Eksempler på denne type gruppevare er utstyr for videokonferanser. *Forskjellig tid og forskjellig sted* representeres av avansert koordineringsteknologi. Ulike konferansesystem og Gruppe Beslutnings Støtte System (GBSS) kan også plasseres i denne delen av matrisen.

### 3.4 Tilpasningsdyktig gruppevareteknologi

Markedssituasjonens turbulente værdag har i den senere tid endret kravene som stilles til gruppevareteknologien. Virksomheter må tilpasses markedssituasjon. For å henge med utviklingen av et marked må virksomhetens struktur endres i takt med markedet. Det påvirker kommunikasjon-, koordinering- og samarbeidsformene. De må derfor utvikles over tid. Det er derfor nødvendig at gruppevareteknologien er tilpasningsdyktig til en organisasjons stadig endrede struktur.

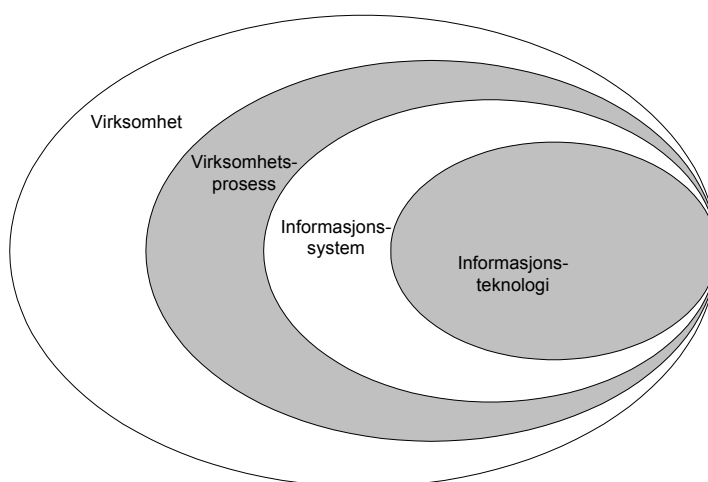
Referanser: [1], [13] og [14]

### 3.5 Informasjonssystem og Virksomhetsprosesser

Et informasjonssystem skal holde orden på informasjonsflyt og informasjonstilgang innenfor et virksomhetsområde. I et systemperspektiv kan det sies at informasjonssystem (IS) er en samling informasjonsmengder og informasjonsprosesser med et vist avhengighetsforhold. Timepris og fakturerte timer på produkt eller tjenester i en konsulentorganisasjon kan være eksempel på informasjonsmengder, mens kalkulasjoner og sammenligninger som gjøres på timepris og fakturerte timer er eksempel på informasjonsprosesser.

Et informasjonssystem betraktes ofte som et system for innhenting, overføring, bearbeiding, lagring, fremhenting og presentasjon av informasjon.

De to overordnede målene for et IS er ofte *styringsformålet* og *kommunikasjonsformålet*. Styringsformålet betyr å frembringe informasjon som kan eller vil benyttes som underlag for planlegging og beslutninger. Kommunikasjonsformålet skal skape kontakt mellom de forskjellige delene av virksomheten og ikke minst mellom virksomheten og omgivelsene. Slike formål knytter IS til de organisatoriske arbeidsoppgaver og gjøremål. Derfor vil et IS ha sin berettigelse kun hvis IS knyttes til bestemte oppgaver eller gjøremål. Oppgavene og gjøremålene utgjør deler av avdelingens virksomhetsprosesser.



**Figur 3: Informasjonssystem i et virksomhetsperspektiv**

Figur 3 posisjonerer et IS i et virksomhetsperspektiv. Posisjoneringen av et IS beskriver IS som et forhold mellom virksomhetsprosesser og informasjonsteknologi. Dette danner grunnlag for følgende definisjon på et IS:

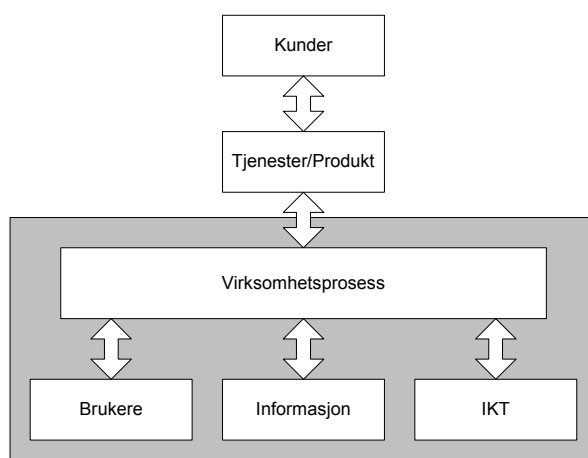
*IS er således systemer som anvender IKT til å innhente, overføre, bearbeide, lagre, fremhente og presentere informasjon som brukes i en eller flere virksomhetsprosesser.*  
[1]

Virksomhetsprosesser blir et sentralt begrep når det gjelder utvikling av nye informasjonssystem. Virksomhetsprosesser defineres ofte som en målbar samling av *strukturerte aktiviteter* eller *gjøremål*, *produkter* eller *resultater*, som er formet med den hensikt å produsere et resultat for intern eller ekstern kunde. Til sammen utgjør de et marked. Eksempel på en virksomhetsprosess kan være faktureringsrutinene for et konsultantselskap.

Virksomheter består av flere gjensidig avhengige virksomhetsprosesser som til sammen utgjør eksistensgrunnlaget for

bedriften. For at en virksomhet skal ha et eksistensgrunnlag er det ikke nok med et sett av virksomhetsprosesser som produserer tilfredsstillende produkt og tjenester. En virksomhet må samhandle med omgivelsene, noe som kalles for virksomhetsomgivelsene. Disse omgivelsene omfatter konkurrenter, kunder, leverandører, politiske myndigheter, rammebetingelser og demografiske, sosiale og økonomiske samfunnsforhold. Sist men ikke minst inngår bedriften selv i virksomhetsomgivelsene.

En *worked-centered analysis* (WCA) er et rammeverk som beskriver forholdet mellom virksomhetsprosesser og det IS som støtter prosessen, se Figur 4. Rammeverkets navn har sin bakgrunn i at det skal hjelpe til med å analysere systemer ved å sette fokus på arbeidsoppgaver og gjøremål som blir utført i en organisasjon.



**Figur 4: Worked-Centered Analysis rameverk**

Rammeverket WCA fokuserer på virksomhetsprosesser, og ikke mot IKT eller IS. Dette er nyttig med tanke på at det er viktig å forstå virksomhetsprosessen før en vurderer et IS som skal støtte prosessen. Sammenhengen mellom en virksomhetsprosess og IS er knyttet mot mengden informasjon som behandles i arbeidsprosessen. Når en prosess er informasjonsintensiv må større deler av prosessen utføres ved hjelp av et IS.

Toveis kommunikasjon mellom elementene i rammeverket prøver å symbolisere at det må være balanse i organisasjonen. En forandring på et av elementene vil forskyve likevekten. Derfor vil en endring i organisasjonen alltid påvirke hele virksomheten som helhet. Når en virksomhetsprosess endres, for eksempel effektiviseres, vil det kreve endringer i behovet for informasjon, bruken av IKT, deltagerens kompetanse. En annen effekt, og ofte målet, er kvalitetsheving på produkter og tjenester.

Referanse [1]



### **3.6 Virksomhetsutvikling**

Markedssituasjon har gjort det nødvendig for virksomheter å tilpasse seg en ny markedssituasjon. Markedssituasjonen er preget av krevende kunder som ønsker rask levering av skreddersydde kvalitetsprodukter til konkurransedyktige priser. Problemet med å levere produkter og tjenester i henhold til gjeldende krav ligger hos mange bedrifter i avdelingenes organisering. De er ofte organisert med det mål for øyet å oppnå funksjonell effektivitet, og ikke for å maksimere nytteverdien for kunden. De fleste virksomheter er etablert rundt prosessene salg, produksjon, innkjøp og økonomi, det vil si at virksomhetene er funksjonsorienterte. Det betyr at oppmerksomheten rettes mot aktivitetene innenfor funksjonsområdene. Mellom de ulike funksjonene eksisterer det "vanntette skott". Derfor ligger det godt til rette for suboptimalisering.

Funksjonsorganiseringen fører til manglende vertikal verdiskapning. Årsaken er at ingen sitter direkte med overordnet ansvar for denne verdiskapningen. Kunder er utelukkende interessert i horisontal verdiskapning. Det skaper problemer for virksomheter. Kunden er interessert i å få tilfredsstilt sine behov gjennom behovstilpassede tjenester og produkt av høy kvalitet og til lav pris.

Dette er rammebetingelser som legger et betydelig press på virksomheter. Slik dagens markedssituasjon er, er de fleste virksomheter preget av krevende individuelle kunder, og en konkurranse som er mer tilspisset en tidligere. Dette fører til at det internt i bedrifter stilles høye krav til omstillingsevne. I dag er ikke kontinuerlig organisasjonsendring unntaket som bekrefter regelen. Det er blitt regelen. Det er under slike omstendigheter at flere virksomheter har oppdaget logikken og verdien i mottoet til Janne Carlzon, *Riv pyramidene!* Tungrodder, hierarkiske organisasjoner er utelukkende egnet til å forsinke verdiskapningsprosessen. Dette betyr på ingen måte at styring og kontroll er blitt avleggs. Det innebærer at det i dag er mulighet for å styre og kontrollere mer smidig og effektivt ved hjelp av moderne IT-systemer.

Referanse [1]

### **3.7 IKT, IS og Virksomhetsutvikling**

Konkurransoeffektivisering handler om tilpasningsevne til konkurranseutsatte omgivelser. Omgivelsene kan være markeder, politiske rammebetingelser, infrastruktur. Konkurransoeffektivisering kan like gjerne være evnen til omstilling, nyskapning og innovasjon. Denne sannheten krever kreative tilnærminger til alt fra produktivitetsforbedringer til utvikling av nye forretningsområder.

De fleste bransjer har i dag blitt møtt av nye krav til endring og omstilling som følge av den økte konkurransen. Omstillingsevne regnes som en nødvendig forutsetning for konkurransedyktighet. Kravene til omstilling er en følge av stadig tilspissing i konkurransen innenfor de fleste bransjer og sektorer. En annen faktor som spiller en

vesentlig rolle når det gjelder kravene til omstillingsevne er at livssyklusen til produkt og tjenester har blitt stadig kortere. Det påvirker tilgjengelig tid til utvikling av nye produkt og tjenester. Hyppig og rask omstilling er blitt en forutsetning for til enhver tid å møte markedets ønsker og behov.

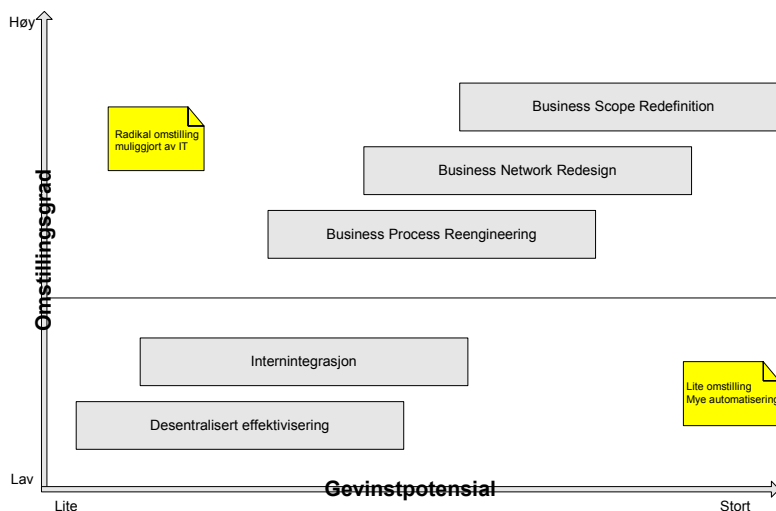
Vedvarende eller økt konkurransevne krever omstilling og virksomhetsutvikling. Endringsevne er på sin side et spørsmål om virksomhetens evne til bevisst, planlagt nyskaping og innovasjon. Her tenkes det på innovasjon som utvikling av produkt og tjenester såvel som arbeids- og virksomhetsprosesser. Kompetanse og teknologi fremstår derfor som sentrale virkemidler uansett innovasjonstype. Videre i oppgaven rettes oppmerksomheten mot teknologibasert omstilling eller om man vil, omstilling med IKT og IS.

Referanse [1]

### **3.8 Venkatramans rammeverk**

Rammeverket illustrerer sammenhengen mellom *gevinstmuligheter* ved bruk av IKT og *omfanget* av omstilling/organisasjonsendring. Det grunnleggende budskapet i rammeverket er at gevinstmulighetene ved bruk av IKT er svært begrensede hvis teknologien utelukkende benyttes til å automatisere eksisterende funksjoner og arbeidsmåter. Gevinstmulighetene øker i takt med evnen til å benytte IKT til å realisere organisatoriske endringer. I rammeverket benyttes fem former for virksomhetsutvikling. De fem formene karakteriseres ved at gevinstmulighetene øker i takt med omfanget av organisatoriske endringer. Det er viktig å være klar over at det er gevinstmulighetene som øker, og ikke de faktiske realiserte gevinstene.

Rammeverket til Venkatraman deles inn i to nivåer av omstillingsgrad. Det laveste nivået, det *evolusjonære*, er det nivået IKT bidrar til forbedring og effektivisering. Nivået preges av minimalt med endringer i sentrale virksomhetsprosesser. Det evolusjonære nivået deles videre inn i to former for omstilling, *Desentralisert effektivisering* og *Internintegrasjon*. Det *revolusjonære* området karakteriseres ved forandringer av virksomheten generelt og/eller sentrale virksomhetsprosesser. Det revolusjonære nivået deles inn i tre former for endring, *Business Process Reengineering (BPR)*, *Business Network Redesign (BNR)* og *Business Scope Redefinition (BSR)*.



Figur 5: Venkatramans rammeverk

### 3.8.1 Det evolusjonære nivå

*Desentralisert effektivisering* tar i bruk IKT i all hovedsak for å automatisere arbeidsprosesser og kostnadseffektivisere. Ansvarlige ledere innenfor adskilte funksjonsområder tar isolerte beslutninger om bruk av IKT i den hensikt å effektivisere sitt funksjonsområde. Det er ofte snakk om relativt enkle systemer. Slike systemer kan være et regnskapssystem, lønnsystem eller et lagringssystem. I andre tilfeller kan det være nyere teknologi som anvendes innenfor adskilte funksjonsområder. Eksempler kan være systemer for ruting av telefonsamtaler og faktureringsinformasjon.

*Internintegrasjon* har som formål å integrere relaterte aktiviteter i virksomheten ved hjelp av integrerte informasjonssystemer og felles databaser. Undersøkelser Venkatraman foretok viser at integrasjon kan bety to ulike ting, *teknisk konnektivitet* og *virksomhetsprosessavhengighet*. Teknisk konnektivitet handler om måten ulike IKT-systemer samvirker på via en felles teknologisk plattform. Virksomhetsprosessavhengighet dreier seg om hvordan ulike arbeids- og virksomhetsprosesser er relatert til hverandre i form av felles datagrunnlag eller sekvensiell avhengighet mellom arbeidsoperasjoner til tross for at de utføres av forskjellige personer innenfor ulike funksjonsområder.

De evolusjonære formene for omstilling gir i utgangspunktet begrensede gevinstmuligheter som følge av "asfaltering av krøtterstier". Det fører vanligvis ikke til synlige endringer i opplevd kunde verdi fordi IKT-løsninger av disse typene er enkle å kopiere. Vegen er imidlertid kort til de revolusjonære formene for endring. Det hele blir et spørsmål om grad av omstilling eller virksomhetsutvikling. På det revolusjonære nivå er det ikke IKT som spiller hovedrollen slik som på det evolusjonære nivå. På det øverste nivå i Venkatramans rammeverk spiller IKT en rolle som en katalysator som tillater

virksomheter å organisere og gjennomføre arbeidsprosesser på en helt ny måte.

### 3.8.2 Det revolusjonære nivå

*Business Process Reengineering* (BPR) handler gjerne om radikal prosessforbedring. Det snakkes gjerne om innovativ fornying av større eller mindre deler av forretningsvirksomheten. BPR kom som en ny bølge ved inngangen til 1990 åra. Med fornying tenkes det på radikal forandring av virksomhetsprosesser, rutiner og måter å utføre arbeid på. Vanligvis tenkes ikke på isolerte forbedringer av enkeltstående virksomhets- og arbeidsprosesser, som for eksempel fakturering, inkasso, lagerstyring eller tekstbehandling. Slike forbedringer blir som regel regnet for kostnadseffektiverende. BPR handler om å erstatte det tradisjonelle funksjonshierarkiet med en mer prosessorientert organisering. Det betyr derfor at BPR for mange bedrifter vil være ensbetydende med å starte på nytt. I virkeligheten og i praksis vil nok det være uaktuelt. Et sitat som beskriver BPR godt er:

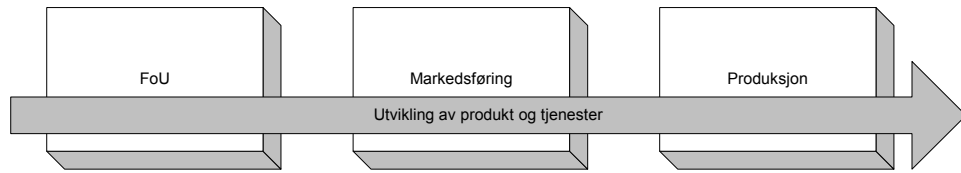
En *fundamental* og *radikal* omforming av virksomhetsprosesser i den hensikt å oppnå *dramatiske* forbedringer som gir målbare resultater på kriterier som kostnad, kvalitet, service og (lede)tid. [1]

*Fundamental* i sammenhengen BPR tar utgangspunkt i grunnleggende spørsmål av typen: Hvorfor gjør vi det her? Hvorfor gjør vi det på den gitte måten? Hensikten med disse grunnleggende spørsmålene er å avdekke aktiviteter, rutiner og arbeidsmåter for om mulig å avdekke irrasjonellitet og ineffektivitet.

*Radikal* betyr å ta problemene ved roten. Målet er å tenke nytt og kreativt med tanke på nye måter å løse arbeidsoppgaver på. Enkelte forskere anbefaler at man ser bort fra eksisterende arbeidsdeling og arbeidsmåter, for på den måten å starte på blanke ark. Den måten for prosessforbedring anses av andre som helt urealistisk i en praktisk situasjon, og at den derfor best egner seg teoretisk.

*Dramatiske* resultatforbedringer er målet ved enhver BPR prosess. Gjennom et BPR prosjekt er det dramatiske forbedringer man er ute etter. Mottoet kan da være "10x, ikke 10%".

*Virksomhetsprosesser* danner utgangspunktet for alle BPR prosjekt. Årsaken er at ledere flest er for lite prosessorienterte. De fokuserer for mye på oppaver, jobber, mennesker og strukturer. Sentrale virksomhetsprosesser er ofte tverrfunksjonelle av natur. Det betyr at de spenner over flere funksjonsområder i virksomheten. Figur 6 illustrerer en tverrfunksjonell virksomhetsprosess.



**Figur 6: Tverrfunksjonell virksomhetsprosess**

BPR har sin opprinnelse fra amerikanske IKT konsulentmiljøer. Det har ført BPR defineres som *omforming av virksomhetsprosesser gjennom kreativ bruk av IKT*. Kreativ utnyttelse av muliggjørende faktorer og katalysatorer fører ofte til gode BPR resultater. Prosessfornying er bare mulig dersom det i et BPR prosjekt lykkes å utnytte IKT og menneskelige ressurser. Selv om IKT spiller en indirekte eller direkte rolle i de fleste BPR prosjekt finnes det mange eksempler gode resultater gjennom kreativ reorganisering av menneskelige ressurser.

*Business Network Redesign (BNR)*, ofte på norsk kalt for Interorganisatorisk samhandling. Her fokuseres det på virksomhetens relasjoner til omgivelsene. Ellers representeres BNR nivå med endringer basert på IKT løsninger for måten produkt, tjenester og ressurser utveksles og deles på mellom aktører i et verdisystem. Ellers er det vanlig oppfatning at BNR mest av alt dreier seg om å utnytte IKT til å effektivisere dataflyten mellom partene i en handelstransaksjon, såkalt Electronic Data Interchange (EDI). Det eliminerer behovet for registrering av samme data flere ganger, samt raskere effektivering av transaksjonene. Venkatraman slår imidlertid fast at BNR er mye mer. Han trekker da frem fire funksjonsområder.

Det første er *transaksjonsbehandling* som er nært knyttet til EDI. Transaksjonsbehandling dreier seg hovedsakelig om utveksling av strukturerte transaksjonsdata. Strukturerte data er ofte forbundet med utveksling av kjøpsordrer, fakturaer, materialister og betalingsanvisninger. Nøkkelen for effektiv transaksjonsbehandling er standardprotokoller for EDI. Det skal sikre at sender og mottaker forstår hverandre. De viktigste fordelene med transaksjonsbehandling er knyttet til administrativ kostnadsbesparing som følge av dataregistrering en gang, eliminering av portoutgifter, og en betydelig reduksjon i papirarbeid og -flyt.

*Lagerstyring* skal bruke IT til å effektivisere vareflyten i produksjonsprosesser der ulike deler av produksjonen utføres av ulike virksomheter. Et stikkord i forbindelse med lagerstyring kan være *concurrent engineering*. Fordelen er de samme som for transaksjonsbehandling, men i tillegg kommer fordelen med forbedret logistisk effekt.

*Prosesskobling* er en utvidelse av de to foregående. Her handler det om å koble sammen flere prosessfaser ved hjelp av lik systemplattform. Poenget er å sette bort deler av produksjons- og distribusjonsfasen til ulike aktører, uten at hovedansvarlig mister oversikten. Også her er *concurrent engineering* et stikkord.

*Kunnskapsutnyttelse* er det siste funksjonsområdet Venkatraman trekker frem som særdeles viktig. Den formen for samhandling dreier seg om systematisk utnyttelse av ekspertkunnskap. Eksempler kan være multimediatekonferanser, der eksperter på tvers av landegrensener og verdensdeler møtes over nettet for å utveksle ideer og erfaringer. Fordelene er at resultatene av profesjonell arbeidsinnsats blir bedre og mer synlig. I tillegg dannes grunnlaget for organisatorisk og individuell læring.

*Business Scope Redefinition* (BSR), ofte på norsk kalt for Redefinert Kjernevirksomhet. Her fokuseres det på virksomhetens relasjoner til omgivelsene. Sentralt står tanker om hvordan IKT kan muliggjøre en mer effektiv samhandling og arbeidsdeling i markedet gjennom etablering av nye organisasjonsløsninger, som for eksempel strategiske allianser, partnerskap, joint ventures og virtuelle virksomheter. Hovedforskjellen mellom BNR og BSR er at IKT på BSR nivå danner grunnlaget for nye forretningsområder. Dette er ofte en konsekvens av IKT løsninger utviklet for omstilling på de lavere nivåene i rammeverket danner såkalte spin offs. Kjernevirksomheten vil da gjennomgå en utvidelse både i dybde og bredde.

Referanser [1] og [15]

### **3.9 Concurrent Engineering**

Concurrent Engineering (CE) [8] er en systematisk måte å integrere/samkjøre design-, produksjons- og produksjonsrelaterte prosesser. CE omfatter også brukerstøtte. CE skal gjøre prosessene mer samordnet eller parallelle. Målet er at utvikleren, produksjonsarbeideren skal bli mer reflekterte i sine handlinger. Sett fra utsiden av produksjonsprosessen, skal de involverte tenke over alle elementer i et produkts eller en tjenestes livssyklus fra konsept, gjennom planleggingsfaser og produksjon til ferdig produkt. Det inkluderer også kvalitetskontroll, kostnadsstyring, produksjonsplanlegging og brukerkrav.

Sentralt i CE er muligheten for utveksling av informasjon mellom ulike personer tilknyttet utviklingen av et produkt eller en tjeneste. Disse vil ha interesse for ulike deler av den informasjonen som finnes i informasjonssystemet. En person som jobber med produksjon vil noen ganger ha bruk for produksjonsdata, og andre ganger kanskje ha et behov for å vite hvorfor ulike valg har blitt gjort under utviklingen.

CE er som BPR og BNR et resultat av endrede krav i markedet [6]. CE bygger på mange gamle prinsipper, men navnet er ikke gammelt.

Teorien søker å løse problemet med at mange av dagens organisasjoner er preget av hierarkisk struktur og fokus på økonomisk fortjeneste. I mange tilfeller har det ført til at kompleksiteten og barrierene mellom gruppene på grunn av tradisjon og intergruppekjennskap har blitt for stor til å tilfredsstille dagens markedskrav.

### **3.10 Integrated Product Development**

Integrated Product Development (IPD) er en filosofi om at systematisk organisering av tverrfunksjonelle arbeidsgrupper, ut fra funksjoner i arbeidsprosessen, skal føre til at nødvendige tiltak for en effektiv og rasjonell arbeidsprosess ivaretas [8].

IPD medfører at design, ingeniørarbeid og produksjon tar plass i de samme omgivelsene. Det skaper nærmere relasjoner mellom ulike arbeidsfunksjoner gjennom livssyklusen til et produkt. IPD organiseringen integrerer alle aktiviteter fra konsept gjennom planleggingsfaser, produksjon til ferdig produkt ved bruk av multifunksjonelle arbeidsgrupper [9]. Det inkluderer kvalitetskontroll, kostnadsstyring, produksjonsplanlegging og brukerkrav. IPD skal samtidig optimalisere et produkt eller en tjeneste og dets produksjons- og støtteprosesser for å møte kostnad- og ytelseskravene. IPD integrasjonen medfører at både ingeniør- og produksjonsfunksjonene i fellesskap blir holdt ansvarlige for måloppnåelse, kostnadsbalanse og bonuser.

#### **3.10.1 Fordeler med CE og IPD**

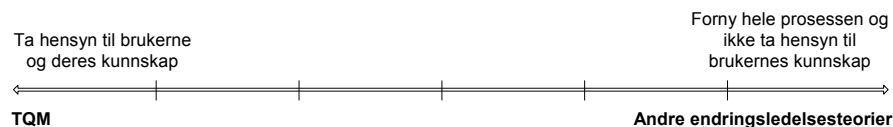
En rapport fra *National Institute of Standards & Technology, Thomas Group Inc. and Institute for Defense Analyses in Business Week*, April 30, 1990 konkluderer med følgende:

Fordelene med CE og IPD er [8]:

- 30% til 70% mindre utviklingstid.
- 65% til 90% mindre "turnover"
- 20% til 90% lavere "time to market"
- 200% til 600% høyere kvalitet
- 20% til 110% høyere "hvitsnipp" produktivitet

### **3.11 Total Quality Management(Total KvalitetsLedelse)**

I Total Quality Management (TQM) blir det fokusert mye på å trekke brukerne inn i utviklingsprosessen som en ressurs. Det sees som en fordel at deres kunnskap om arbeidssituasjonen og arbeidsprosessen blir ivaretatt under kartleggingen av arbeidsflyten og utviklingsfasen. Hos mange andre av endringsledelsesteoriene blir det ikke tatt nødvendig hensyn til brukerne og deres kunnskap. Ofte blir eksisterende prosesser forkastet, og helt nye prosesser blir utarbeidet. Det betyr at en helt ny arbeidsflyt blir definert på grunnlag av utviklerens oppfatninger av situasjonen.



Figur 7: TQM i forhold til andre endringsledelsesteorier

TQM er et ledelseskonsept som i all hovedsak er utviklet i Japan, men med betydningsfulle bidrag fra USA. TQM er på norsk ofte kalt for Total Kvalitetsledelse (TK). TQM er en ledelsesform som tar sikte på å tilfredsstille kundeforventninger gjennom å engasjere alle ansatte i en prosess for kontinuerlig forbedring. Fra arbeidsgiversiden forutsettes det at en tilrettelegger for at den enkelte medarbeider skal kunne bli *motivert* for å delta i en slik prosess, at en utvikler *evne* til deltagelse og ikke minst: Det må gis *mulighet* for slik medvirkning. Samtidig forventes fra arbeidstakersiden at den enkelte medarbeider internaliserer sentrale elementer knyttet til virksomhetens visjon, overordnede mål og grunnleggende strategi. På den måten skal arbeidstakeren delta aktivt i realiseringen av disse grunnleggende elementer i en virksomhet, og at det knyttes positive følelser til en slik målrealisering.

Kvalitetsledelse innebærer en gjennomgående endringsprosess for å bedre kvaliteten på tjenester og produkt som produseres for eksterne såvel som interne kunder. Det betyr at TQM er en kontinuerlig prosess som må gjennomsyre hele virksomheten/avdelingen. Målet er resultatforbedring ved å bruke tilgjengelige ressurser mer rasjonelt og effektivt for å imøtekomme eksterne og interne behov. En viktig forskjell på kvalitetsledelse i forhold til andre endringsledelsesteorier er at de ansatte står i fokus. Det anses som svært viktig at de ansatte, som er de som skal stå for kvaliteten på de ferdige produktene, tas med i endringsprosessen. På den måten er TQM et konsept som gjør kvalitetskontroll til et felles ansvar for alle i organisasjonen.

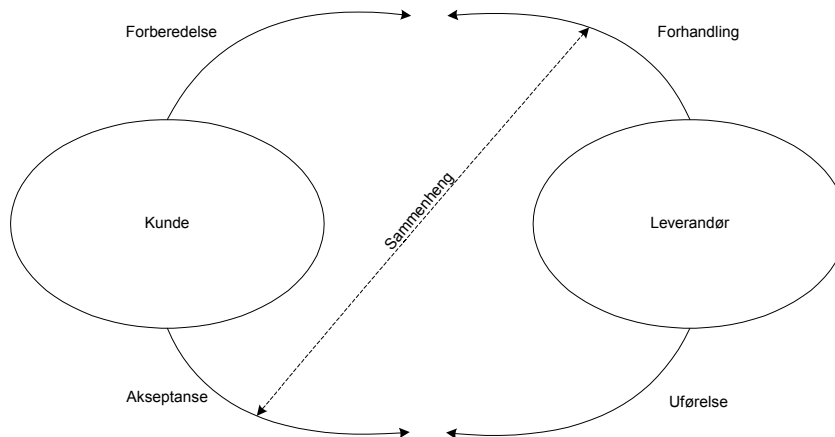
Det grunnleggende i TQM filosofien er at ressurser utnyttes best hvis organisasjonens funksjonsmåte utvikles kontinuerlig for å møte eksterne og interne krav. Den er derfor forskjellig fra innovasjonskulturen, som i større grad bærer preg av at ressurser mobiliseres når det oppstår kritiske problemer. TQM handler om å flytte ressurser fra feilretting, kontroll og "brannslukking" til forebygging av feil. For å belyse TQM nærmere er følgende sitat passende:

Kvalitet er samsvar med forutbestemte krav, ikke godt eller dårlig. Dette innebærer at det må utarbeides klare og entydige bedriftsinterne krav og spesifikasjoner gjennom fokus på kundens behov. Dette tvinger frem en nødvendig markedsorientering [2]

Fire sentrale begrep som til sammen utgjør den generelle forståelsen av god ledelse beskrives nærmere. Forståelsen av hvor viktig disse delene er, øker mulighetene for vellykkede prosjekt. *Forberedelse* er kundens forarbeid før han henvender seg til leverandører med sine



ønsker og krav til tjenester eller produkt. *Forhandlinger* omfatter anbudsrunder og valg av leverandør. Her forhandles det om pris, kravspesifikasjon, leveringstid og rammebetingelser. *Utførelse* er den jobben leverandør legger ned fra oppstart til avslutning av prosjektet. *Akseptanse* er i det tjenesten eller produktet leveres til oppdragsgiver. Akseptansen er verdien kunden ser i produktet i ettertid. Sammenhengen mellom forhandlingsfasen og akseptansfasen avdekker hvorvidt kunden har opplevd kvalitet eller ikke, med andre ord god ledelse.



**Figur 8: TQM som livssyklus**

I Figur 8 viser den stiplede linjen forholdet mellom kundens behov i utgangspunktet (fra forhandlingene) og den verdi kunden oppfatter i ettertid (akseptansen). Denne sammenhengen vil da avdekke hvorvidt kunden har opplevd kvalitet eller ikke.

TQM er ledelse og tilrettelegging av kontinuerlige forbedringsprosesser på alle nivå i bedriften. Det viktigste er å legge vekt på verdiskapning, samarbeidsevne og vilje til å bruke kunnskap i praksis.

### 3.11.1 Konflikter mellom samarbeidsmodellen og TQM

*Den norske samarbeidsmodellen* er preget av en sterk grad av konsensus, ved at partene i mange saker har søkt å samarbeide om felles utfordringer fremfor å kjøre konflikter grunnet i ulike interesser. Hensikten med dette har vært å gjøre bedriftene mer konkurransedyktige gjennom økt involvering og medinnflytelse til de ansatte.

Fra fagbevegelsens side har det vært ønskelig at engasjementet skal skje med grunnlag i formaliserte ordninger hvor de ansattes rettigheter er lov- og avtalefestet. Det er blitt bygget opp et sett med reguleringer som skal sikre en institusjonell basis for de enkelte ansattes deltagelse og engasjement. Derfor kan en innføring av TQM utfordre samarbeidsmodellen.

For *ledelsen* representerer TQM en utfordring på en rekke felter. En stor utfordring ligger i ledelsens forhold til de ansatte når det gjelder kompetanseutvikling og de ansattes medvirkning. TQM konseptet legger stor vekt på at de ansatte trekkes inn i forbedringsarbeidet.

Samtidig ligger det i konseptet en forpliktelse på ledelsen i å utvikle medarbeidernes kompetanse, slik at de er i stand til å utøve denne medvirkningen på en konstruktiv måte for bedriften. Sett opp i mot den norske samarbeidsmodellen samsvarer dette med eksisterende lovverk og avtaleverk. Til tross for de sterke samarbeidstradisjonene i norsk arbeidsliv, er det ingen tvil om at visse områder av bedriftens beslutninger har vært forbeholdt ledelsen. Et eksempel på dette er strategiplanleggingen der det har vært tradisjon for at ledelsen har monopol. I TQM bedrifter hvor man satses på medvirkning og kompetanseutvikling kan et slikt område på sikt bli utfordret av en mer kompetent og påvirkningsvillig arbeidsstokk.

For *ansatte* representerer TQM på den ene siden større mulighet for selvrealisering og innflytelse over egen arbeidsplass. På den andre siden fører TQM til økte krav til de ansatte. TQM konseptet forutsetter at den enkelte ansatte utvikler sine samarbeidsevner, sin kompetanse, sin fleksibilitet, sitt ansvar og sine evner til å behandle kunderelasjoner. Økte krav til ansvar og oppfølging kan medføre økt stressfaktor og øke psykososiale arbeidsbelastninger for de ansatte som opplever TQM som belastende på sin arbeidshverdag.

*Hovedutfordringen* for både ledelse, ansatte og fagbevegelsen må være å utvikle organisasjonsmodeller som kan integrere det beste fra både TQM konseptet og den norske samarbeidsmodellen på en måte som gir økt konkurransekraft og gode arbeidsplasser.

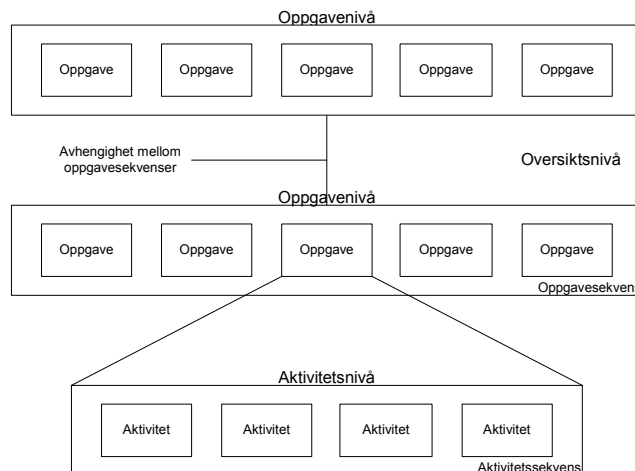
### 3.11.2 Arbeidsflyt og arbeidsflytsystem

Arbeidsflyt er automatisering av arbeidsprosesser som er styrt av prosedyrer. Et arbeidsflytsystem er et proaktivt system for å styre saker gjennom forhåndsdefinerte behandlingsprosesser og kan oppfattes som produksjonsstyring. Arbeidsflytsystem er en egen kategori gruppevare, og brukes til å støtte grupper i løsning av felles oppgaver.

Definisjonen på arbeidsflyt og arbeidsflytsystem varierer mye fra hvilket behov og egenskap som ligger i den aktuelle arbeidsflyt. Derfor finnes det uttallige definisjoner på arbeidsflyt og et arbeidsflytsystem. Begrepene *arbeidsflyt* og *arbeidsflytsystem* defineres ut fra case *faktureringsprosess for konsulenter*. Velger å definere en arbeidsflyt slik:

*Arbeidsflyt* er en forhåndsdefinert sekvens av arbeidsoppgaver som er avhengige av, og påvirker hverandre, slik at det bidrar til at organisasjonen når sine mål [4].

Det er vanlig å skille mellom tre nivåer i en arbeidsflyt. De tre nivåene er *aktivitetsnivået*, *oppgavenivået* og *oversiktsnivået*. Aktivitetsnivået og oppgavenivået beskriver sekvensen med henholdsvis aktiviteter og oppgaver. Oversiktsnivået angir avhengigheten mellom ulike oppgavesekvenser, se Figur 9. Disse tre nivåene er gjensidig avhengig av hverandre. Det betyr at hvis det oppstår endringer eller feil i et av disse nivåene, vil arbeidsflyten feile. Arbeidsflyten baseres på de oppgavene som skal utføres og det er derfor helt avgjørende å få med alle forhold som spiller inn for at oppgavene skal bli utført tilfredsstillende.



**Figur 9: Aktivitets-, oppgave- og oversiktsnivå satt i sammenheng**

Når det designes og utvikles et IKT system for styring av en arbeidsprosess dannes et arbeidsflytsystem.

*Arbeidsflytsystem* er et IKT støttet system som leverer og tar vare på informasjon om status og fremdrift i prosjekt til ledelsen, slik at de kan kontrollere og styre arbeidet som blir utført.[4]

Et Arbeidsflytsystem kalles på engelsk for Workflow Management System (WfMS). For at et utvikling av et WfMS skal bli vellykket er det viktig at aktivitetssekvensen og oppgavesekvensen blir nøye kartlagt. Spesielt viktig er det å få oversikt over hvordan aktiviteter i en prosess, og hvordan de "beveger" seg fra prosess til prosess. Denne kartleggingen er alfa omega for at et systemutviklingsprosjekt for et WfMS skal bli vellykket.

Utvikling og bruk av systemer for arbeidsflyt har ført til mye diskusjon internt i miljøene som forsker på CSCW. Spesielt har forskerne vært opptatt av hvordan arbeidsflytsystemene kategoriserer, formaliserer og automatiserer flytene. Denne interessen har blitt besvart av empiriske studier som bekrefter at brukerne av systemene synes de er vanskelige i bruk.

På tross av dette har arbeidsflytsystemer stått frem som en løsning på problemene med å koordinere hendelser, gjenstander og ansatte. Dette løses ved å redusere kompleksiteten på koordineringen ned i tre fundamentale steg.

*Første* steget er at arbeidsflytsystemer forsøker å løse dette ved å redusere de grunnleggende formalitetene ved hjelp av å kategorisere prosessene. Under kategorisering av en arbeidsprosess brytes prosessen ned i tre elementer. Elementene representerer aktiviteter (aktivitetsnivået), oppgaver (oppgavenivået) og brukerregler. Når kategoriseringen er gjennomført kan forbindelsene mellom de forskjellige komponentene i en arbeidsprosess formelt bli definert.

*Andre* steg er derfor å presist beskrive avhengighetene mellom de ulike komponentene i et arbeidsflytsystem ved hjelp av formalisme. Det definerer forholdet mellom spesifiserte komponenter (oversiktsnivået).

*Tredje* steg på vegen til et WfMS er *Automatiseringen*. Når kartleggingsprosessen er ferdig kan kategoriseringen og formaliseringen implementeres i et arbeidsflytsystem. Automatiseringsprosessen bruker formaliseringen som grunnlag for automatisering av hele eller deler av en prosess. Det sikrer at en prosess benytter gjeldende regler.

Utviklingen av forståelige og akseptable modeller av arbeidsprosesser er avgjørende for å utvikle et vellykket arbeidsflytsystem. Utfordringen ligger i å finne måter å støtte individet på, slik at brukerne finner et slikt system nyttig i bruk. En modell for en arbeidsflyt konstrueres fra kategorisering og formalisering gjort tidligere i implementeringsprosessen [3].

Mange virksomheter er kontinuerlig på utkikk etter måter å redusere kostnader og tiden assosiert med produksjon av produkt og tjenester. WfMS sees på som en måte å kontrollere disse utfordringene. Dessuten finnes en rekke styrelsesmetoder som forklarer bruken av informasjonsteknologi, inkludert arbeidsflytsystemer, som støtte til arbeidsprosesser. Det kan vises til gode eksempler på at arbeidsflytsystemer har vært suksessfulle. Spesielt har Configuration Management Systems (CMS) vist seg å være effektive [3].

For forskere som er opptatt av å forstå og forbedre kvaliteten på arbeid, har de tre stegene kategorisering, formalisering og automatisering ledet til mange spørsmål. Sentrale spørsmål for forskerne har vært:

- Hvem disse systemene tjener?
- Hvor tilfredsstillende formaliseringen tar vare på arbeidet som modelleres?
- Hvordan verktøyet påvirker brukermiljøet?
- Hvor tilpasningsdyktig systemet er til endrede forutsetninger?

Spørsmålene har ført til utvikling av en ny type arbeidsflyt, adaptiv arbeidsflyt.

### **3.12 Adaptiv arbeidsflyt**

Trenden med at forutsetningene stadig endres har gjort at forskere på CSCW har flyttet sitt interesseområde over fra statiske til mer dynamiske system. Trenden har ført til et behov for adaptivitet. Dynamiske system går under terminologien adaptive system. I adaptive system er flyten fremdeles den sentrale aktivitet, men det er vanlig å gjøre flyten tilpasningsdyktig i forhold til endringer i interne og eksterne faktorer.

Et selskap består av mange ulike deler som til sammen danner en virksomhet. Et arbeidsflytsystem er en viktig komponent. Endrede forutsetninger vil påvirke de ulike komponentene.

En stor mangel ved Workflow Management System (WfMS) er at de antar ideelle prosesser. Den ideelle prosess er godt strukturert og predefinert. Det forekommer aldri. Derfor må et arbeidsflytsystem ta høyde for at virksomhetsprosesser har avvik. Da må et WfMS være tilpasningsdyktig til endrede forutsetninger fra såvel interne som eksterne krefter.

Automatisering av arbeidsprosesser fører ofte til at prosessen blir statisk. Det betyr at arbeidsflyten har liten tilpasningsevne til endrede forutsetninger. Forutsetningene kan endres som følge av [10]:

- Forandring i omgivelsene

De fleste ingeniørorganisasjoner er under konstant utvikling, de er dynamiske. Når forretningsvirksomheten er under konstant forandring og konkurransen er verdensomspennende, er radikal redesign og optimalisering av eksisterende prosesser uunnværlig. Redesign og optimalisering er et virkemiddel for å oppnå bedre ytelse og effektivitet, i de raskt endrede omgivelsene. I tillegg er det behov for endring av forretningsprosessene mellom de radikale endringene i virksomhetsprosessene. Det gjør det mer hensiktsmessig med konstant evolusjon av forretningsprosessen isteden for radikale endringer av virksomhetsprosesser.

- Teknologisk utvikling

IKT system er utsatt for endring i forventningene til datastøtten som tilbys. Årsaken ligger i den teknologiske utviklingen. Teknologisk nyvinning leder ofte til at systemet må rekonfigureres. Bakgrunnen er at software komponenter erstattes og oppdateres som følge av nye komponenter og komponent grensesnitt.

- Tilfeldige avvik fra arbeidsflyt

Tilfeldige avvik oppstår ofte i en arbeidsflyt. Det krever situasjonsbetinget tilpasning av arbeidsflyten. Situasjonsbetingede avvik fra den ideelle prosess kan kategoriseres som følger [10]:

- Dynamisk dannelse

I enkelte tilfeller er det nesten praktisk umulig å utvikle en komplett arbeidsflytmodell i mangel av en komplett spesifisering av prosessen. Derfor er dynamisk tilpasning av arbeidsflyten nødvendig run time.

- Bruker engasjement

Av og til må de involverte i en arbeidsflyt ta styring over arbeidsflyten. De involverte må justere prosessen under påvirkning av eksterne hendelser/omstendigheter. Justering kan komme som følge av personlige tilpasninger i flyten.

- Uforutsigbare hendelser

Uforutsette hendelser oppstår på grunn av interne og eksterne påvirkninger, innblanding av brukere, time out etc. Disse hendelsene må håndteres etter beste måte av arbeidsflytsystemet. Det er viktig for å bevare en eventuell kommunikasjon mellom flere prosesser, slik at samvirkende prosesser ikke påvirkes av uforutsigbare hendelser i en annen prosess.

- Feilsituasjoner

Systemfeil, tilgangskonflikt for systemkomponenter og feilaktige utførelser forårsaker feilsituasjoner. Det skaper vanskeligheter i utførelsen av den automatiserte arbeidsflyten. Mekanismer som håndterer feilsituasjoner er derfor viktig for å oppnå en problemfri arbeidsflyt.

### 3.12.1 Klassifisering av adaptiv arbeidsflyt

Et klassifikasjonssystem av arbeidsflyt er hensiktsmessig for å finne løsninger som håndterer tilpasninger i en arbeidsflyt. De ulike tilpasningskategoriene kategoriseres som følger [10]

- Prosessnivå

Her er det snakk om tilpasninger til arbeidsflytens modell og ulike oppgaver. Modellen må tilpasses endrede rutiner, mens oppgavene må tilpasses tilfeldige avvik. Utviklingen av modellen kan komme som følge av store prosessforandringer og kommer ofte som følge av BPR.

- Ressursnivå

Her må arbeidsflyten tilpasses endringer og omprioriteringer av ressurser som inngår i prosessen. Denne kategorien tilpasning spesifiseres videre [10].

- Organisasjons relatert

Det betyr endringer i organisasjonens struktur og ressurser. Det kan være endringer i sammensetning og personell.

- Data relatert

Data og datastrukturer kan endres under arbeidsflytprosessen. Vanligvis kan data som ikke er i bruk av WfMS bli forandret uavhengig og aksessert av andre applikasjoner. Men hvis en arbeidsflyt er avhengig av eksistens av noen gitte data må prosessen bli tilpasset endringene i dataene. Det forutsetter at forandringene må gjøres kjent for WfMS.

- Infrastrukturnivå

På grunn av at prosessen utvikles for å møte de gjeldende krav i organisasjonen og teknologisk utvikling må arbeidsflytsystemet raskt tilpasses det modifiserte forretningsmiljøet og teknologiske nyvinninger. Derfor må IKT systemet som støtter opp om prosessens infrastruktur bli etablert på en måte som muliggjør effektiv tilpasning til forandringer i infrastrukturen.

### 3.12.2 Suksessfaktorer for utvikling av et adaptivt arbeidsflytsystem

Først vil jeg peke på suksessfaktorer ved et arbeidsflytsystem. Deretter kommer jeg med mine egne kriterier for å øke suksessgraden ved innføring av et adaptivt WfMS. Det er fire hovedelementer det fokuseres på når man analyserer suksessgraden ved innføring av et WfMS [3].

- Utviklerne må forstå og aksepterte den modellerte arbeidsmodell
- Det sørger for forståelige og brukbare representasjoner
- Riktig jobb prosessautomatiseres
- Bedriften støtter opp omkring prosessen på alle områder

For adaptive system er det fem elementer som påvirker suksessen.

- Systemet må kunne tilpasses gjeldende markedssituasjon

- Systemet må kunne tilpasses gjeldende organisasjonsstruktur
- Systemet må kunne tilpasses hurtig og enkelt
- Systemets tilpasninger må ikke påvirke hvordan brukeren oppfatter systemet
- Brukerne må være innforstått med at systemet "lever"

### 3.12.3 *Arbeidsflytteknologier versus manuelle arbeidsprosesser*

Arbeidsflytteknologier er ikke universelt suksessfulle. En ting er å oppfylle suksessfaktorene (se kap. 3.12.2) som et minstekrav, men det er også mange andre faktorer som spiller inn.

Arbeidsflytssystemer, som alle andre komplekse system, kan feile og være en suksess samtidig. Årsaken er at et system som innføres kan fungere utmerket under gitte omstendigheter, mens det under andre omstendigheter på samme prosess kan føre til forsinkelser og feilhandling. Årsaken er at systemene for arbeidsstøtte ofte er store kommersielle produkt. På den måten mistes spesialisering i verktøyet.

Problemet kan ofte ligge nede på menneskelig nivå. Det forsøkes ofte løst ved å lage dynamiske system der det er mulighet for å personalisere verktøyet. På den måten håper de kommersielle produktene å løse dette problemet. Størst sikkerhet for at et prosesstyringssystem skal fungere fås ved å utvikle et spesialisert system for gitte prosess. Det er ofte tidkrevende og svært dyrt. Suksessgarantier kan heller ikke gis her, da det kan være grunnleggende antagelser som er feil, men den reelle sjansen er høyere.

Manuelle arbeidsprosesser kan også ha tilpasningsproblemer. Når endringer i godt innarbeide prosesser blir foretatt må personellet som deltar i prosessen være innforstått med endringene. Det er ikke alltid tilfelle. De involverte føler at ledelsen ikke hører på deres synspunkter. De føler seg forbigått av ledelsen. De involverte føler at ledelsen trer nye rutiner ned over hodene på dem. Ofte forstår ikke prosessens involverte ledelsens argumenter for å endre på godt innarbeidede flyter. Problemet ligger ofte i at "arbeidere på golvet" er i opposisjon til ledelsen. Løsningen på det er å trekke inn de involverte i utviklingsprosessen. De vil da føle at de blir hørt. Denne ledelsesprofilen følger prinsippene i TQM.

Problemene som oppstår under endring av manuelle prosesser gjelder også for WfMS. Derfor vil et WfMS medføre både tekniske og sosiotekniske utfordringer for ledelsen. Utfordringene kan løses ved TQM.

### 3.12.4 *Adaptiv arbeidsflyt i perspektiv til en faktureringsprosess*

En faktureringsprosess kan påvirkes av interne og eksterne faktorer. *Forutsetninger, ressurstilgang, organisasjon og partnerskap* er elementer som påvirkes av omgivelsene. Det kan endre rutinene for hvordan faktureringen foregår eller utføres. Det vil påvirke



arbeidsflytprosessen, og dermed endre forutsetningene for hvordan arbeidsflytsystemet håndterer faktureringsprosessen. Det åpner for bruk av et adaptivt system. Et adaptivt WfMS i faktureringsprosessen gjør at prosessen kan være under kontinuerlig forandring og forbedring for å møte endringer i omgivelsene. Det gjør at bedriften til en hver tid har en optimal prosess for fakturering. Det kan gi virksomheten et lite konkurransefortrinn i forhold til sine konkurrenter. Konkurrentene sitter kanskje med et statisk system som ikke er adaptivt til omgivelsene. Det gjør systemet lite hensiktsmessig og effektiv.

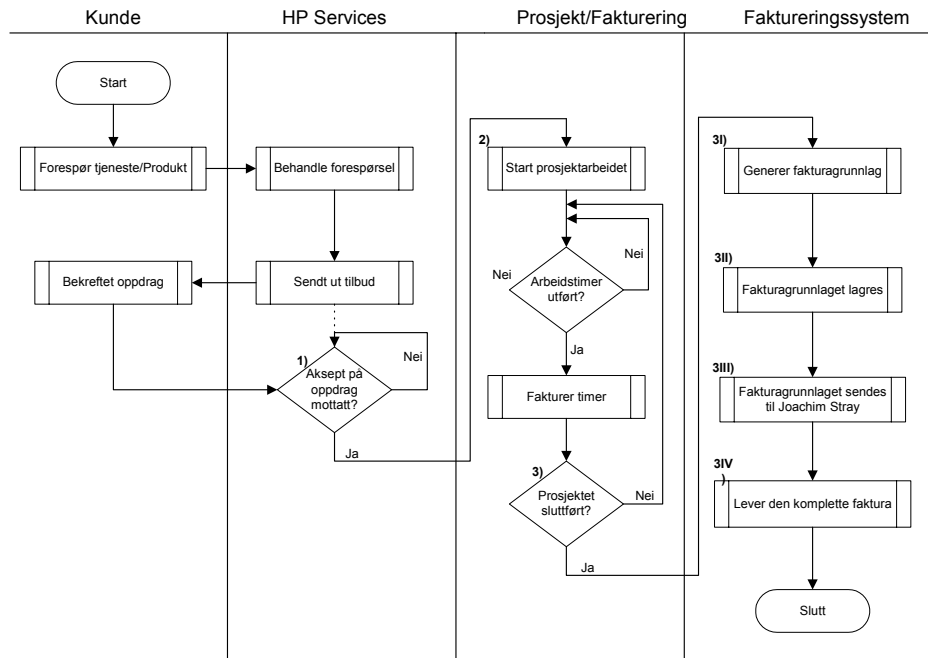
På den andre siden kan et system som er under kontinuerlig forandring og forbedring skape usikkerhet hos brukerne. Da vil adaptiviteten virke mot sin hensikt. Usikkerheten hos brukerne kan minimaliseres med god og riktig informasjon. Det er viktig at informasjon om endringene i faktureringsrutinene kommer i god tid. Det kan oppnås ved at brukerne trekkes inn i den kontinuerlige endringsprosessen. Endringsprosessen vil da gjennomføres av TQM konseptet. Det vil gjøre overgangene til de nye rutinene enklere og mer feilfri. Endringene vil være akseptert av brukerne. Mindre feil i overgangsfasen øker kvaliteten på faktureringen.

I tillegg må et adaptivt system på best mulig måte ivareta brukernes behov for å tilfredsstille brukerne. Det er nødvendig for å få tilpasningene til å gå så problemfritt som mulig. Det er derfor viktig at brukerne trekkes inn tidlig i utviklings- og designprosessen for å gjøre systemet så godt som mulig for brukerne.

## 4 Prosessforbedring

### 4.1 Nåværende Rutine for fakturering i HP

Figur 10 viser hvordan et prosjektets livssyklus er i dag. Prosessen skal forbedres og automatiseres.



Figur 10: Nåværende faktureringsrutine

Faktureringsrutinen foregår i dag etter følgende "regler": Figur 10 inneholder referanser til punktene i "regelsamlingen".

1. Kunden skal gi aksept på oppdraget skriftlig. Oppdrag som er gitt muntlig bekreftes fra HP eller fra kunden med e-post. Bekreftelsen skal inneholde en enkel beskrivelse av oppdraget, når det skal bli utført og informasjon om hvordan fakturering foregår.
2. Start prosjektet.
3. Når oppdraget er utført skal fakturagrunnlaget genereres og en bekreftelse skal sendes kunden.
  - I. Fakturagrunnlag genereres ved å legge data inn i *MSOinput.xls* regnearket. Alle felt skal fylles ut. Merge dataene inn i word dokumentet *MSOfakturagrunnlag.doc*
  - II. Lagre det ferdige dokumentet med følgende navn: <kundens vanlige forkortelse>\_<dine initialer>\_<dato for faktura>.doc. Eks *CMAX\_JS\_020300.doc*
  - III. Send dokumentet til Joachim
  - IV. Skriv ut dokumentet  
Stift dokumentet og den originale forespørselen sammen

Legg dokumentet i posthyllen til Siw Linda.

#### **4.2 Evaluering av nåværende faktureringsløsning**

Beskrivelsen av nåværende faktureringsprosess tilsier at det er rom for forbedringer, automatisering og kvalitetshevelse. De manuelle elementene i prosessen er kvalitetsnedsettende.

Faktureringsprosessen er i dag sekvensiell, i stor grad manuell og lite strukturert, se Figur 10. Sekvensiell og manuell betyr at prosessen er lite effektiv og rasjonell. Utfordringer ligger i å gjøre faktureringen mer effektiv og strukturert. At løsningen virker å være lite strukturert gjenspeiler organiseringen av faktureringsprosessen. Mange av oppgavene og aktivitetene har stort potensial for å bli optimaliserte ved en automatisering. Det kan oppnås med gode tekniske løsninger. Systemløsninger som optimaliserer, effektiviserer og strukturerer arbeidsprosesser er WfMS. Slike systemløsninger kan kjøpes eller utvikles spesielt for en gitt prosess.

#### **4.3 Strategi for prosessforbedring**

Målet med den nye prosessen er at konsulentene skal ta mer ansvar for egen fakturering. Ansvaret for at faktureringen utføres korrekt, nøyaktig og i riktig tid flyttes over på konsulenten. Den nye løsningen skal motivere konsulentene til kontinuerlig fakturering under prosjektene. Konsulentene er brukerne som skal bruke faktureringsystemet hver dag, og deres aksept er viktig for å få en vellykket innføring. Det stiller krav til brukervennlighet og aksept. Systemet må være enkelt og intuitivt i bruk.

For at en slik endringsprosess skal være gjennomførbart må nåværende faktureringsprosess forbedres, rasjonaliseres og effektiviseres. Da må først og fremst prosessen forenkles, og bygges på konsulentenes erfaringer med det gamle systemet. Hvis dette neglisjeres risikerer ledelsen at systemet virker mot sin hensikt.

TQM kan representere uttrykket "frihet under ansvar". Det harmonerer til målet om at konsulentene skal ta mer ansvar for egen fakturering. TQM er en ledelsesform som tilfredsstillende forventningene gjennom å engasjere alle ansatte i en prosess for kontinuerlig forbedring. Det innebærer at brukerne, som er de som kjenner prosessene best av alle, skal trekkes inn i utviklingsprosessen. Det hindrer at innføringen oppleves som om ledelsen trer helt nye systemer ned over hodene på brukerne. Det gjør innføringen og problematikken rundt enklere. Brukerne kjenner svakhetene med dagens prosess og hvilke kvaliteter som er nødvendige for å optimalisere prosessen. Det sikrer at det nye systemet har de kvaliteter som er nødvendig for å oppnå et godt fungerende system på prosessnivå.

På den andre siden kan hensynet til konsulentenes ønske og meninger føre til "asfaltering av gamle krøtterstier". Det vil ikke være særlig innovativt. Resultatet kan bli at ønsket forbedring ikke oppnås. Det er viktig å være bevisst på dette.

TQM innebærer gjennomgående endringsprosess for å bedre kvaliteten på prosessen. Skal et TQM utviklingskonsept lykkes, forventes det fra arbeidsgiversiden at det tilrettelegges slik at den enkelte medarbeider skal bli motivert til å delta i en slik kontinuerlig endringsprosess. Det forventes også at den ansatte skal utvikle evne til deltagelse. Her er det viktig å være klar over at det må gis mulighet for medvirkning.

Samtidig forventes det at den ansatte tar aktivt del i utvikling av virksomhetens eller avdelingens visjon, overordnede mål og grunnleggende strategi. På den måten håper ledelsen å oppnå at det knyttes positive følelser til en slik målrealisering.

Problemet med TQM som ledelsesprofil er at det stilles økte krav til ansvar og oppfølging fra brukerne. Det kan føre økt stressfaktor og øke psykososiale arbeidsbelastninger for de ansatte som opplever TQM som belastende på sin arbeidshverdag.

For ledelsen kan TQM oppfattes som en trussel, selv om samarbeidstradisjonene som finnes i norsk arbeidsliv er sterke, er det ingen tvil om at visse områder av bedriftens beslutninger har vært forbeholdt ledelsen. Strategiplanleggingen er et slikt område. Der har det vært tradisjon for at ledelsen har monopol. I TQM bedrifter hvor man satser på medvirkning og kompetanseutvikling kan et slikt område på sikt bli utfordret av en mer kompetent og påvirkningsvillig arbeidsstokk.

Når markedssituasjonen er turbulent krever det at kommunikasjonen og samarbeidet er av ypperste kvalitet internt på avdelingen og mellom konsulenter og prosjektledere. Det er nødvendig for å møte konkurransen på best mulig måte. Da er gruppevare teknologi basert på CSCW en løsning som vil tilfredsstillende behovene til kommunikasjon og samarbeid. For å gjøre kommunikasjon og samarbeid så godt som mulig må systemet gi tilbakemelding med informasjon som kan brukes til beslutningsstøtte. Det kan være viktig får å ta riktige strategiske beslutninger. Det krever at systemet gir konsulent informasjon for beslutningsstøtte under faktureringsprosessen. Støtten brukes av konsulenten for å registrere strategisk informasjon. Det danner grunnlag for beslutningsstøtte hos ledelsen. For å få kommunikasjon og samarbeid så effektivt som mulig, må beslutningsstøtten som gis konsulent og prosjektleder være tilgjengelig som endelig fakturainformasjon på et prosjekt. Det krever at timeregistreringen som gjøres på et prosjekt, faktureres prosjektet i samme øyeblikk. Det betyr at timeregistrering og prosjektfakturering foregår samordnet/i parallell, såkalt CE. Følgende sitat støtter opp om samordning/parallellkjøring av time registreringsprosessen og endelig prosjekt fakturering:

I et prosessperspektiv er det samordning/parallellkjøring av fasene, såkalt Concurrent Engineering som kan forbedres. Dette forutsetter bedre koordinering og samordning av selve prosessen, samt nye samarbeids og kontraktsformer [7].

Sitatet viser at valget av gruppevareteknologi basert på CSCW og CE kan gi gode resultater. CSCW tar hånd om kommunikasjon og samordning, mens CE representerer strategien for å forbedre prosessen.

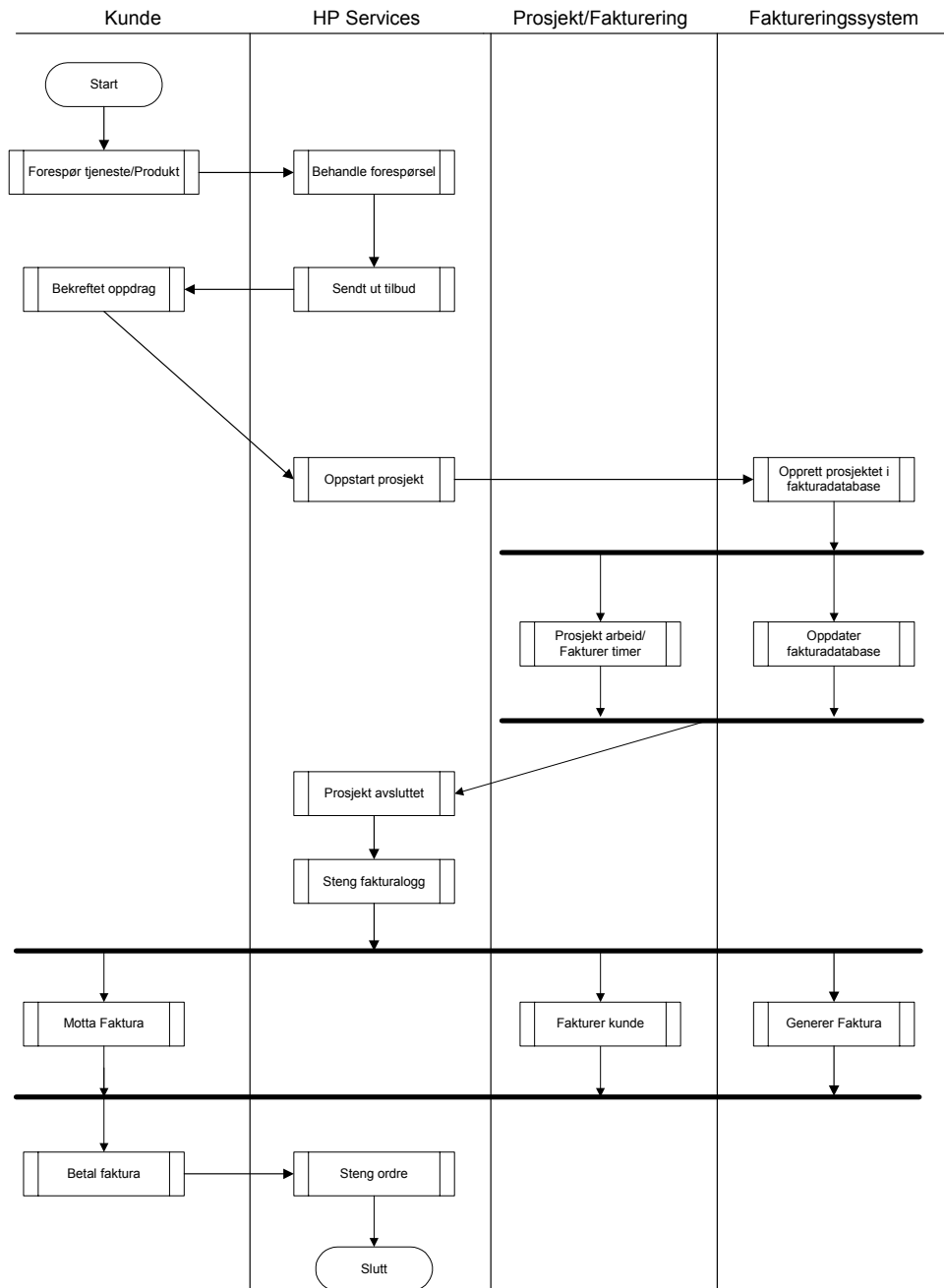
Når endelig fakturering av et prosjekt går samordnet/parallellkjørt med konsulentenes timeregistrering oppstår mulighet for feilfakturering av et prosjekt. På den måten kan gale beslutninger bli tatt. Samtidig er det rimelig å anta at prosjektleder skulle reagere hvis store avvik fra forventet fakturerte timer på et prosjekt forekommer.

På den måten kan store feil oppdages. Problemer oppstår når småfeil registreres. De vil ofte være vanskelig, eller helt umulig å oppdage. Småfeil må unngås, da "mange bekker små, gjøre en stor å". Det er derfor konkurransemessig uheldig at slike feil oppstår. Det er her TQM som ledelsesprofil skal søke å være holdningsskapende, slik at registreringen gjennomføres så nøyaktig som mulig. Videre skal TQM ivareta brukernes ønsker under utvikling og design av systemet. Det skal sikre at systemet innehar de kvalitetene som konsulentene forventer. De vet "hvor skoen trykker", og kan derfor sikre at utviklingen og designet av systemet på best mulig måte vil unngå feilfakturering.

Valget av TQM som ledelse- og utviklingsprofil og gruppevareteknologi basert på CSCW som teknologisk løsning harmonerer godt til det som står sentralt i prosessforbedring med CE. CE omfatter kvalitetskontroll, kostnadsstyring, produksjonsplanlegging og brukerkrav. Produksjon av faktureringsgrunnlag må kvalitetssikres på best mulig måte. Det søkes løst med kombinasjonen mellom CE og TQM. CE har tidligere vist seg å være effektivt i forbindelse med kostnadsstyring og produksjonsplanlegging. Kostnadsstyring og produksjonsplanlegging er det som representerer beslutningsstøtten hos konsulent og prosjektleder.

#### **4.4 Forslag til prosessforbedring med arbeidsflytstøtte**

Prosessforbedringen presenteres i Figur 11. Den presenterer mitt forslag til et prosjektets livssyklus og hvordan fakturering foregår etter den nye modellen. Forklare kort den nye prosessen. Deretter kommenterer jeg hvilke elementer som arbeidsflytstøtten skal automatisere.



Figur 11: Forslag til prosessforbedring

Flyten er i stor grad samordnet/parallell, mer effektiv og pålitelig. Samordnet/parallell oppnås når flere av oppgavene og aktivitetene utføres samordnet eller samtidig. Det påvirker også effektiviteten til hele arbeidsprosessen. Pålitelig henpeiler på at den er mer automatisert. Flytelementer som posisjonerer seg mellom to horisontale linjer utføres samordnet/i parallell, eller alternative prosessvalg. Konsulentenes fakturering av timer og oppdatering i fakturadatabasen foregår samordnet/i parallell med timeregistreringen. På den måten genereres fakturagrunnlaget kontinuerlig etter hvert som konsulentene fakturerer timer. Det gjør

det mulig med beslutningsstøtte. Når et prosjekt "stenges", genereres automatisk faktura til kunde.

Slik frigis personalressurser når et prosjektets fakturagrunnlag skal genereres. Personalressursene kan brukes på administrative gjøremål under oppstart av nye prosjekt, under prosjektarbeid og ved prosjektavslutning. Med den nye modellen kan derfor starten av nye prosjekt fremskyndes. Totalt vil den nye arbeidsflyten gjøre livssyklusen til et prosjekt kortere. Resultatet er en mer effektiv prosjektorganisasjon.

Når fakturagrunnlaget genereres automatisk vil ledelsen miste litt oversikt. Ledelsen vil i mindre grad være direkte involvert i faktureringen, mens mer av ansvaret legges over på konsulentene. En slik ledelsesprofil følger TQM og konfliktene som kan oppstå i forbindelse med den. For enkelte i ledelsen kan dette være frustrerende, da de mister sin innflytelse på fakturagrunnlaget. Derfor er det viktig med god formell koordinering og støtte rundt de økonomiske transaksjonene (se kap. 8.2).

Automatiseringen tas hånd om av et WfMS. Det er utviklet en prototyp for å demonstrere hvordan et WfMS kan løse automatiseringen i faktureringsprosessen. Registrering av fakturagrunnlag foregår automatisk ved at timer konsulenten registrer automatisk oppdateres i fakturadatabasen. Slik unngås en manuell prosess med å registrere timer i fakturadatabasen ved hver månedsavslutning. Når registreringen er kontinuerlig vil fakturadatabasen inneholde den "sanne" faktureringsstatusen på et prosjekt. På den måten er det mulig til enhver tid å hente ut gjeldene fakturagrunnlag på et prosjekt. På den måten tilrettelegges det for at arbeidsflytssystemet kan tilby beslutningsstøtte. Når et prosjekt stenges vil systemet automatisk generere det totale fakturagrunnlaget til kunden.

## **5 Teknologi**

Det ligger føringer fra HP Services om at systemet skal utvikles på hyllevare. Leverandør av utviklingsplattformen skal være Microsoft. HP Services ønsker å designe systemet rundt Office 2000 og Outlook 2000. Det er likevel nødvendig å undersøke kommende versjon av Office pakken, Ms Office XP, og dens teknologiløsninger og nye muligheter. Databasen kan fritt velges fra Microsofts mange databaseløsninger.

### **5.1 Microsoft Office 2000**

Applikasjonene i Office 2000 kan skreddersys til kundens behov. For å lette utviklingsarbeidet har utviklerne tilgang til en stor plattform av predefinert funksjonalitet som det kan designes løsninger over. På den måten kan kundens behov tilpasses og integreres med andre applikasjoner for å levere sømløs og godt tilpassede løsninger. Kunden vil få sine løsninger utviklet over en plattform som er godt kjent blant brukerne. Det gjør innføringen enklere og øker aksepten blant brukerne. Produktpakken Office 2000 er den mest utbredte pakken for Office applikasjoner. Det gjør det enkelt å tilpasse applikasjonen til flere organisasjoner.

Referanse [16]

### **5.2 Microsoft Office 2000 Developer**

Microsoft Office 2000 Developer er utviklet spesielt for profesjonelle utviklere som utvikler nye applikasjoner under Office 2000 med Microsoft Visual Basic for Applications. Microsoft Office 2000 Developer inneholder et produktivetsverktøy som profesjonelle utviklere behøver for å utvikle løsninger raskere og et integrasjonsverktøy for å gjøre løsningsarbeidet med data og nettet sømløst. Den inneholder dessuten et forvaltningsverktøy for enkelt å gjøre applikasjonen som er bygget med Office tilgjengelig.

Å bygge løsninger på toppen av produktivetsapplikasjoner fører til en raskere utviklingssyklus og fornøyde brukere. Årsaken er at organisasjonelle data kan leveres direkte i produktivetsapplikasjonene som brukerne benytter hver dag. På grunn av disse fordelene bygger mer enn 2,5 millioner utviklere sine løsninger på Microsoft Office. Dette gjør Office til den nest største plattformen målt i utviklere, etter Microsoft Windows.

Referanse [16]



### 5.3 Microsoft Outlook 2000

Outlook 2000 er meldings- og samarbeidsklienten i Ms Office. Den organiserer elektronisk post, kalender, avtaler, kontakter og oppgavelister. Outlook 2000 forbedrer kommunikasjon og samarbeid i hele organisasjonen.

### 5.4 Utviklingsmetoder i Microsoft Outlook

Det finnes tre former for utvikling i Ms Outlook. De tre formene for utvikling og design er Outlook forms, Office Document Forms og Visual Basic for Application.

#### 5.4.1 Outlook Forms

Outlookforms er beregnet for løsninger der få endringer i forhold til standard Outlook forms er nødvendig. Det finnes seks Outlook forms som kan skreddersys til kundens behov. Disse er; *Appointment, Contact, Journal Entery, Meeting Request <Hidden>, Message, Post, Task* og *Task Request <Hidden>*. Bruk av disse "krever" at funksjonaliteten som utvikles passer inn i en av de forhåndsdefinerte forms og at det er små forandringer i forhold til de seks standard forms.

#### 5.4.2 Office Document Forms

Likheten med de innebygde Outlook Forms er stor, men i tillegg har Office Document Forms funksjonalitet som gjør dem svært interessante. Med Outlook kan man utvikle Office Forms som inneholder funksjonaliteten og programmerbarheten til et Word-, Excel worksheet-, Excel chart dokument eller PowerPoint presentasjon. Det er mulig å bruke Microsoft Visual Basic Scripting Edition (VBScript) på "innsiden" av Outlook og Microsoft Visual Basic for Applications til å gi Office dokumentet den nødvendige funksjonalitet. Typisk vil Outlook Office Forms bli brukt i situasjoner der en løsning skal rute Office dokumenter til en bestemt person via E-post eller der et Office dokument skal deles (post) med andre. Dokumentet vil da bli liggende i en public folder.

#### 5.4.3 Visual Basic for Application (VBA)

Det er enklere å utvide Office applikasjoners funksjonalitet ved bruk av VBA. Årsaken er at utviklingen ikke er knyttet til en av de seks standard Outlook Forms.

VBA er en kraftig utviklingsteknologi for rask tilpassing og integrasjon av standard programmer med eksisterende data og systemer.

VBA tilbyr et sofistikert sett med programmeringsverktøy basert på utviklingssystemet Microsoft Visual Basic, verdens mest populære utviklingsplattform, som utviklere kan bruke til å få mest mulig ut av Office programmene. VBA gjør at kundene kan kjøpe programvare i

butikken og deretter tilpasse den til sine bestemte forretningsprosesser, isteden for å bygge opp programmene fra bunnen. Det betyr, spart tid og penger, redusert risiko, fordi det er enkelt å dra nytte av programmeringsmulighetene av Office applikasjonen.

VBA gir et fullstendig og integrert utviklingsmiljø, IDE (Integrated Development Environment), som inneholder de samme elementene som utviklere kjenner fra bruk av Microsoft Visual Basic, inkludert et prosjektvindu, et egenskapsvindu og feilsøkingsverktøy. Visual Basic for Application inneholder også støtte for Microsoft Forms for oppretting av tilpassede dialogbokser, og ActiveX-kontroller for rask bygging av grafisk brukergrensesnitt (GUI).

VBA integreres direkte i et vertsprogram, og tilbyr fordelene med rask ytelse gjennom prosessen (opptil 200 ganger raskere enn andre frittstående utviklingsverktøy), tett integrasjon med vertsprogrammet (kode bak dokumenter, celler osv.) og muligheten til å bygge løsninger uten å bruke tilleggsverktøy.

Programmer som omfatter VBA, kan skreddersys for å dekke bestemte forretningsmessige behov. Denne klassen programmer gir utviklere mulighet til raskt å bygge løsninger som krever mindre sluttbrukeropplæring. Når det gjelder Management Information Systems (MIS), betyr muligheten for tilpassing at løsningene kan utvikles raskt og brukes enkelt med minimalt vedlikehold. I en bransje som er vant til to års innkjøringstid for nye programmer og høye kostnader på sluttbrukeropplæring, representerer slike løsninger vesentlige fordeler i form av større kapitalavkastning og redusert ledetid.

Referanse [17]

### **5.5 Microsoft Office XP**

Den nyeste oppgraderingen av kontorsuiten Office, Office XP, retter seg hovedsakelig mot bedriftsmarkedet. Mange mener at systemkravene til Office XP ligger litt vel høyt. Systemkravene er de samme som for Windows 2000, noe som gjør at mange brukere vil ha for lite minne på sin maskin til å kjøre programmet.

Office XP tilbyr mange nye automatiske funksjoner. Det har vært en del kritikk fra brukere om at de automatiske funksjonene lar seg verken slå av eller kontrollere. I Office XP skal det være enkelt å slå av all uønsket automatikk.

Office XP redefinerer forholdet mellom mennesker og programvare ved å tilby en ny og bedre måte å arbeide på. Office XP tilbyr blant mange nye funksjoner blant annet:

- En funksjon kalt SmartTags gjenkjenner informasjon mens du taster, og kan tilby forskjellig type assistanse
- Nye forståelige feilmeldinger

- Task Pane (oppgaverute) funksjonalitet
- Funksjon for tekst til tale generering
- Funksjon for tale til tekst generering
- Egenskaper som gjør programvaren mer stabil og bedre utstyrt til å bevare dokumenter ved systemkrasj

### 5.5.1 SmartTags

SmartTags er små knapper som settes inn i et dokument i forbindelse med ulike gjøremål. Noen smartTags vises hele tiden, andre kommer bare til syne når musen beveges over et område der en SmartTag er satt. Formålet med SmartTags er først og fremst å forenkle arbeidet med å redigere og utforme dokumentene. SmartTags dukker opp i en rekke sammenhenger. SmartTags kan kommunisere med adresseboken i Outlook. En SmartTag kan eksempelvis automatisk hente inn en persons adresse straks du skriver navnet, eller vise når du sist hadde en avtale med vedkommende.

Tekstbehandleren i Office XP har lært seg å kjenne igjen personnavn. Lar du musen fare over et personnavn, kommer en SmartTag til syne med mange forhåndsdefinerte muligheter.

SmartTags kan benyttes av avanserte brukere eller organisasjoner for å tilrettelegge Office-applikasjonene. De er COM objekter som støtter XML. En SmartTag kan kopieres og programmeres videre på.

Videre tenkes SmartTags brukt som integrasjonsverktøy mellom Office og Internett. Et nettsted kan tilby SmartTags som legger tabellinformasjon rett inn i et regneark, for eksempel salgsstatistikk, børskurs og så videre. Lenken mellom det lokale regnearket og nettstedet opprettholdes, slik at informasjonen holdes ajour.

### 5.5.2 Forståelige feilmeldinger

Kryptiske feilmeldinger i Office er erstattet av gode forklaringer på hendelser som er skjedd, slik at feilmeldingsrapporteringen er kraftig forbedret. De gode og forståelige forklaringene preger alle feilmeldinger.

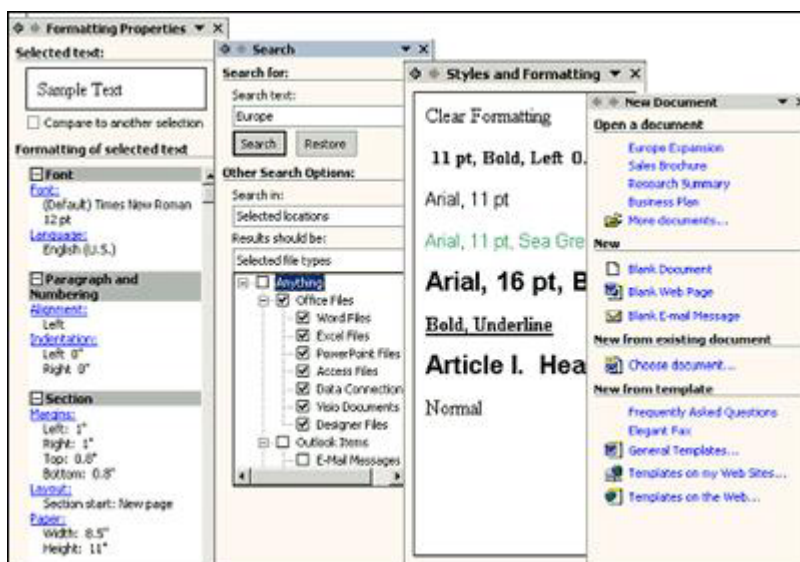
### 5.5.3 Task Pane

Office XP tilbyr Task Pane for å styre gjøremål fra en rute til høyre for dokumentet. Task Pane vises til høyre for dokumentet i Office applikasjoners skjermbilde. Formålet er å samle valgmuligheter som hører til ditt gjøremål, og som applikasjonen antar skal utføres på grunnlag av den kommandoen som nettopp ble gitt.

Noen eksempel er "lim inn", "formater tekst", "søk", "velg stilmal", "åpne dokument". Bildet nedenfor viser hvordan det kan se ut i Excel når noe skal limes inn, og utklippstavlen inneholder flere elementer.

Month	Sales Target	Sales Goal	% of Target
January	\$8,258	\$8,312	111.60%
February	\$7,888	\$8,250	107.88%
March	\$5,588	\$7,164	123.23%
April	\$8,588	\$7,500	96.87%
May	\$18,090	\$14,587	125.24%
June	\$12,250	\$13,432	108.19%
July	\$14,850	\$12,888	94.28%
August	\$11,750	\$10,878	91.94%
September	\$18,800	\$14,587	123.96%
October	\$8,888	\$8,125	101.37%
November	\$8,888	\$8,000	98.73%
December	\$8,888	\$10,458	123.49%
Total	\$118,198	\$125,949	107.75%

Forbedringen er ganske åpenbar. Office 2000 lar også utklippstavlen ta vare på flere elementer, men det må stoles på hukommelsen for å vite hva som er hva. I Office XP vises klart hvilken applikasjon snittet er hentet fra, og du gis noen linjer med tekst for å gjøre det lettere å identifisere snittet enda klarere. Nedenfor vises andre eksempler på oppgaveruter, klippet fra Excel, PowerPoint og Word:



Meningen er å gjøre det enklere å bruke applikasjonene, og samtidig minne om muligheter som gjerne ikke er så lett tilgjengelig fra menyene.

#### 5.5.4 Tekst til tale

En funksjon for stemmegenerering som går fra *tekst til tale*. På den måte er det mulig å få lest opp hva du har skrevet. Denne funksjonen er nyttig hvis det skrives inn store summer i for eksempel Excel, og det skal sjekkes at det er skrevet korrekt siffer.

#### 5.5.5 Tale til tekst

En funksjon for tekstgenerering som går fra *tale til tekst*. Hvis datamaskinen kobles til telefonen eller mikrofon kan for eksempel et intervju bli nedskrevet i løpet av telefonsamtalen, og deretter kan

utskrift av intervjuet tas. Denne funksjonen er kunstig intelligent, det kreves nemlig at brukeren trener den opp.

Referanse [18]

#### 5.5.6 *Ms Office 2000 vs. Ms Office XP*

Føringer fra HP gjør at Office 2000 skal benyttes som utviklingsplattform. Det er en programpakke som vil være dominerende en tid fremover. Neste generasjons kontorpakke, Office XP, vil være kontorpakken de fleste velger å oppgradere til. Tiden fra Office XP kommer i salg til den er allment utbredt utgjør et generasjonsskifte.

Når vi vet at et systemet utvikles over tid, burde det kanskje vært satset på Office XP. Årsaken er at et system alltid tar tid å utvikle, slik at utviklingen vil komme midt inn i generasjonsskiftet. Resultatet kan bli at faktureringsystemet er ferdig utviklet når generasjonsskiftet er over. Det kunne derfor vært på sin plass å starte utviklingen av systemet rundt denne fremtidige plattformen, som representerer helt ny teknologi, slik at systemet fra første dag er kompatibelt med neste generasjons Office brukere. Nå risikeres at det satses på teknologi som er gammel når systemet er ferdig utviklet. Synet på valg av teknologi forsterkes når vi vet at Office XP byr på ny og spennende funksjonalitet. Mye av denne funksjonaliteten ser jeg for meg at det kunne vært utnyttet i systemet.

På den andre siden kan manglende dokumentasjon av Office XP gjøre utvikling på plattformen vanskelig. Tilgangen på dokumentasjon kan gjøre utviklingsarbeidet tungt for utvikleren. Det vil likevel fortløpende mot lanseringen, og en stund etter lanseringen, komme mye dokumentasjon som er nyttig for Office utvikleren. Spørsmålet er bare om dokumentasjonen som utvikleren behøver der og da kommer til riktig tid. Denne problematikken gjør det vanskelig å utvikle rundt Office XP. Med andre ord en større utfordring for utvikleren. Det er rimelig å anta at det er enklere å utvikle et system rundt Office 2000, for senere å konvertere applikasjonskoden til Office XP.

#### 5.5.7 *Valg av utviklingsmetode i Outlook*

Outlook Forms brukes når løsningen som utvikles har få endringer i forhold til standard Outlook Forms. Det betyr at funksjonaliteten må passe inn i standard Outlook Forms. Det gir begrenset mulighet til å legge inn spesiell funksjonalitet. Det er mulig å utvikle et system for prosessforbedring etter spesifikasjonene fra HP Services i Outlook Forms. System bygd på Outlook Forms vil ha begrensede muligheter for å videreutvikles med ny og avansert funksjonalitet. Det begrenser systemets potensial.

På den andre siden kan det være fornuftig å gjøre utviklingen av prototyp på enkleste måte, og ikke tenke for mye på fremtidig videreutvikling. Når det er sagt kan mye nyttig funksjonalitet oppnås

ved bruk av riktig Outlook Forms. Det reduserer utviklingstiden for applikasjonen. Dessuten kan mye feil unngås i denne grunnleggende funksjonaliteten. Årsaken er at funksjonaliteten er godt utprøvd.

Office Document Forms har stor likhet med Outlook Forms. Den store forskjellen er at Office Document Forms tilbyr mulighet for utvikling av Office Forms som inneholder funksjonaliteten og programmerbarheten til et Word-, Excel worksheet-, Excel chart- dokument og PowerPoint presentasjon. Her er det mulig å kombinere bruken av VBScript og VBA. VBScript brukes på innsiden av Outlook Forms, mens VBA brukes for å gi Office dokumentet de ønskede egenskapene.

Det åpner mulighetene for å legge inn mye nyttig funksjonalitet fra Office applikasjonene i systemet. På den måten kan de forhåndsprogrammerte Office Forms og programmeringsmuligheten til et Office dokument utnyttes. Det gjør det enkelt å legge til nyttig funksjonalitet fra Office dokumentene på en enkel måte.

Denne løsningen kan virke som en kompromissløsning når det gjelder valg av verktøy for å oppnå ønsket funksjonalitet. Valget kan virke å bli "mellom barken og veden". Tanken bak utsagnet er at det gis og tas fra begge sider. Valget av utviklingsverktøy blir derfor midt i mellom to verktøy. Spørsmålet er om det påvirker kvaliteten på det som utvikles. Ender valget på utviklingsverktøy på Office Document Forms, kan det kanskje være like greit å utvikle den ønskede funksjonaliteten i VBA.

VBA er den kraftigste utviklingsmetoden i Outlook. VBA tilbyr funksjonalitet for rask tilpasning og integrasjon av standard programmer med eksisterende data og datasystemer. VBA tilbyr den mest avanserte funksjonalitet for å skreddersy applikasjoner til ønskede behov. Videre tilbyr VBA et fullstendig og integrert IDE miljø. En annen stor fordel er at VBA støtter bruk av ActiveX kontroller for raskt og effektivt å kunne designe avansert GUI.

Det store spørsmålet ved valg av VBA som utviklingsverktøy, er om det "skytes spurv med kanoner". Harmonerer ikke valget av modell/teori, bruk/behov og teknologi kan systemet komme i ubalanse. Det er også en høyere inngangsterskel for bruk av VBA, enn de to andre formene for utvikling i Outlook. Det kan gi en tregere start av utviklingen, men vil gi større mulighet for avansert funksjonalitet.

Når det gjelder Office Forms og Office Document Forms mister utvikleren mye av kontrollen på applikasjonen. Årsaken er at mye av koden ligger skjult i predefinert programmeringsfunksjonalitet. Det kan gjøre det vanskelig å feilsøke om det oppstår en feil. Utvikling i VBA gir programutvikleren full kontroll over egen kode. Det vil gjøre feilsøking enklere. Det kan spares mye tid under feilsøking ved bruk av VBA. VBA er et komplett utviklingsverktøy, som gjør at utvikleren har tilgang på mye nyttig funksjonalitet til hjelp i utviklingen og det administrative. VBA står frem som den komplette utviklingsmetode i Ms Outlook. HPFaktura vil bli utviklet i VBA.

## 5.6 Database

Når løsningen skal designes og utvikles rundt Office 2000 og Outlook 2000 er Access 2000 og Exchange Server 2000 nærliggende valg.

## 5.7 Microsoft Access

Med Microsoft Access 2000 er det enkelt å hente informasjonen du trenger. Det er et kraftig verktøy som hjelper til med å ordne og dele databasen slik at teamet ditt kan ta bedre avgjørelser. I Access 2000 har utviklere valget mellom to Database Management system (DBMS), ofte kalt en databasemotor. Det enkleste DBMS er Jet Data Engine, kjent fra tidligere versjoner av Access. Den andre databasemotoren er Microsoft Data Engine (MSDE).

### 5.7.1 Jet Data Engine (JDE)

Er det enkleste DBMS som mottar og håndterer data fra bruker og systemdatabasen, og lagrer data i den. JDE er dataadministratoren som styrer alle spørringer og lagringer mot en Access datafil. JDE er utviklet og optimalisert for bruk i små datasystemer, slik som enbruker og små arbeidsgrupper.

### 5.7.2 Microsoft Data Engine (MSDE)

Er det avanserte DBMS som mottar og håndterer data fra bruker og systemdatabasen, og lagrer data i den. MSDE er dataadministratoren som styrer alle spørringer og lagringer mot et Access dataprojekt. MSDE er fullt kompatibel med Microsoft SQL Server 7.0. Det fører kraften i høy databaseytelse inn i det velkjente Access miljøet. På den måten kan det dras nytte av skalerbar SQL Server teknologi i Access. Det kan brukes til å opprette databaser som skal støtte forretningsprosesser som krever høy pålitelighet og skalerbarhet. På den måten kan systemet utvides med en databaseskalering som passer til de varierende forretningsbehovene.

MSDE benytter klient server arkitektur. På den måten kan det opprettes klient server programmer i Access grensesnittet. Det fører til økt brukervennlighet.

### 5.7.3 Jet Data Engine (JDE) versus Microsoft Data Engine (MSDE)

Hvis bruken av databasen skal støtte opp om forretningsprosesser eller kunderelasjoner i en virksomhet er MSDE å foretrekke. Selv om bruken i første omgang ikke er på virksomhetsnivå vil bruk av Access i "Front end" med MSDE i "Back end" sikre at skalerbarheten er til stede når virksomheten vokser og/eller når systemet videreutvikles. Virksomhetsapplikasjoner krever skalerbarhet, adaptivitet, sikkerhet og robusthet. De tre kravene kan tilfredsstilles med MSDE, men ikke med Jet Data Engine.

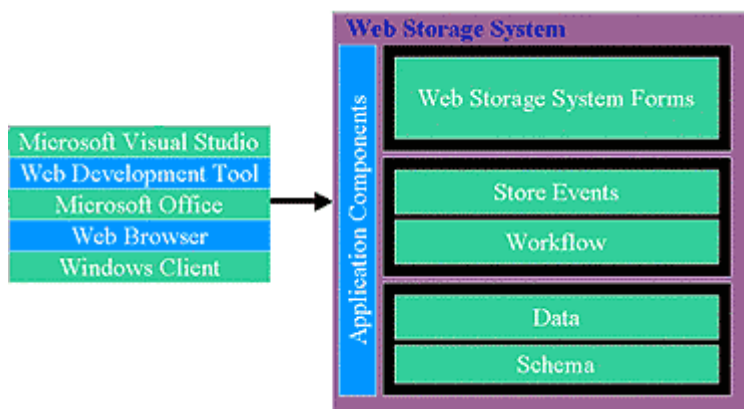
Referanse: [19]

## 5.8 Microsoft Exchange 2000

Microsoft Exchange 2000 server er utviklet for å møte dagens behov innenfor gruppevare. Løsninger designet på denne plattformen skal fungere for alle virksomheter, uansett størrelser. Exchange 2000, og dets klient, Microsoft Outlook 2000, er ment å være en skalerbar løsning for gruppevare implementasjon. Exchange organiserer data i private (typisk e-post) eller fellesmapper.

### 5.8.1 Web Storage System

I Microsoft Exchange 2000 introduseres en ny teknologi, Web Storage System. Denne teknologien er tiltenkt utviklere som bygger kommunikasjons- og samarbeidsløsninger i sine applikasjoner. Web Storage System gjør det mulig for programmereren å utvikle de ferdige integrerte løsningene, slik at eksisterende applikasjoner kan bli utvidet med kommunikasjon- og samarbeidsløsninger eller det kan bygges helt nye applikasjoner for gruppevare tilpasset brukerens behov. Se Figur 12 for presentasjon av klienter og utviklingsverktøy.



**Figur 12: Presentasjon av klienter og utviklingsverktøy som kan aksessere et Web Storage System**

## 5.9 Ms Data Engine versus Ms Exchange 2000

Det er et krav at prototyp skal utvikles på Ms Outlook 2000 plattformen. Da er et nærliggende valg er å benytte Ms Exchange 2000 Server som databaseløsning. Den tilfredstiller kravene til skalerbarhet, adaptivitet, sikkerhet og robusthet. Dessuten inneholder Exchange 2000 Web Storage System. Det gjør det mulig for programmereren å utvikle de ferdige integrerte løsningene, slik at eksisterende applikasjoner kan bli utvidet med kommunikasjon- og samarbeidsløsninger. På den måten tilfredstiller Ms Exchange mye av kravene til databaseløsning.

Videre ønsket HP Services å få vurdert muligheter og begrensninger ved en senere integrering med eksisterende faktureringssystem. Det stiller krav om at databasen må være enkel å tilpasse Ms SQL Server. Årsaken er at de fleste store databasene er kompatible med den. Det åpner for bruk av en database som er kompatibel med Ms SQL server.



Da er det mest nærliggende å benytte en database som er 100% kompatibel med Ms SQL Server 7.0. Det gjør MSDE til beste alternativ som database. Det åpner muligheten for en senere integrering med eksisterende system. Databasen som systemet utvikles for å kommunisere med er derfor Ms Data Engine (MSDE) Den tilfredsstillende kravene til skalerbarhet, adaptivitet, sikkerhet og robusthet.

### 6 Databasemodell og design av database

Før design og implementering av databasen tar til, er det nødvendig å gjennomføre en behovsanalyse for å finne egnet modell og design.

#### 6.1 Valg av databasemodell

Presenterer tre scenarier som beskriver tre datamodeller. Datamodellene ble presentert HP Services, slik at de kunne velge den de følte var best egnet. Mine kommentarer gis umiddelbart etter hvert av modellbeskrivelsene. HP Services' kommentarer er gjengitt i kursiv for valgte modell.

1. Loggene er grunnenhetene i systemet. Det betyr at databasemodellen designes rundt loggene. Hver enkelt konsulent er selv ansvarlig for å opprette loggen, levere logg til riktig tid, og riktig ført, uten oppfølging fra overordnede eller systemet. Loggen føres etter gjeldende regler, og det er opp til konsulent å følge disse.

Dette er den enkleste modellen. Hensikten med modellen er å skape et databasesystem som ligner dagens system, men automatisert. Styrken til modellen er at den er svært lik nåværende system, og av den grunn reduseres innføringsproblematikken.

Datamodellen fokuserer lite på grunnleggende inntektskilder i organisasjonen. Når systemet fokuserer lite på konsulent, prosjekt og kunde kan brukerne føle at systemet ikke tjener brukerne. Det kan påvirke konsulentenes holdninger til systemet.

2. Databasen inneholder til enhver tid alle nåværende, som tidligere kunder, i et flatt kunderegister. I et flatt kunderegister ligger det ingen prioriteringer av kunde på størrelse og pågående prosjekt. Kundene legges fortløpende inn i databasen. Kunden registreres ved kontraktsinngåelse. På den måten slipper konsulent å registrere kundeopplysninger ved opprettelse av logg. Her er prosjekt og konsulent grunnleggende entiteter.

Ved å flytte fokuset bort fra fakturering av timer, og over til grunnlaget for inntektene, prosjekt og konsulent, er intensjonen å skape en datamodell, og et system, som setter prosjekt og konsulent i sentrum. Modellen støtter derfor opp om TQM. Modellen harmonerer derfor godt til valgt strategi for prosessforbedring. Når prosjekt og konsulenter settes i fokus kan det virke positivt på faktureringsgraden per konsulent. På den måten kan det likevel oppnås at faktureringsgraden øker. Årsaken er at trivselsfaktoren kan bedres når konsulentene føler at fokuset ikke er faktureringsevnen, men deres velvære i forbindelse med prosjektgjennomføring. Konsekvensen kan bli

økt trivsel. Bedret trivsel fører ofte til økt faktureringsgrad. Trekket dette langt kan det argumenteres med at trivsel reduserer bedriftens "turnover".

Usikkerheten er om fokuset som velges vil føre til økt fakturering. I verste fall går faktureringsgraden ned, stikk i strid med intensjonen.

*Tror det er denne som blir mest riktig.*

3. Prosjekt er grunnleggende entitet, og loggene utgjør underentiteter. Det betyr at prosjekt er overbygning på loggene. Det knyttes en ny logg til hvert prosjekt ved ny faktureringsmåned. Loggene vil inneholde aktivitetsregistrering. Se på loggene som knyttes opp mot prosjektene som en kjede med fakturainformasjon. Prosjektentitetene knytter til seg metainformasjon. Metainformasjon kan være kundeinformasjon fra databasen, informasjon om konsulenter tilknyttet prosjektet osv.

I denne modellen er prosjekt og logger grunnleggende entiteter. Modellen konsentrerer seg omkring entitetene som danner inntektskildene, prosjekt og fakturering (logger). Målet er å sette prosjekt og fakturering i sentrum. Fokuset i datamodellen skal bevisstgjøre konsulenten i at loggføring er en viktig del av prosjektgjennomføring.

Svakheten er at "den viktigste" kilden til inntekter, konsulenten, ikke er tatt med i databasemodellen. At konsulenten ikke er en grunnleggende entitet i modellen kan føre til at hans interesser overses. Modellen tilfredsstiller derfor ikke valgt strategi for prosessforbedring fullt ut. Konsulentene som til syvende og sist danner inntektene kan føle at systemet er til for ledelsen, og ikke dem. Ved denne modellen kan derfor fokuset risikere og bli feil, sett fra konsulentens side. Når den viktigste inntektskilden føler at fokuset blir for mye på prosjekt og loggføring kan det nye systemet risikere å bli en hodepine for ledelsen.

### 6.1.1 Oppsummering databasemodeller

Databasemodellene har tre forskjellige innfallsvinkler. Den første modellen er enkel og fokuserer på loggen som grunnleggende enhet. Denne modellen ligger nærmest dagens, men er noe mer automatisert. Har vel en heller tvilsom gevinst, bortsett fra automatiseringen. Den andre datamodellen har prosjekt og konsulent som to sentrale entiteter. Modellen flytter fokus bort fra fakturering, og fokuserer på prosjekt og konsulent. Den tredje har prosjekt og logg som sentrale entiteter. Her er fremdeles fakturering en viktig del av modellen, men i tillegg trekkes prosjekt inn som en sentral enhet.

### 6.2 Design av databasemodell

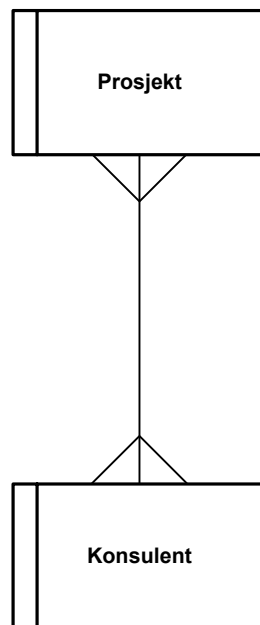
Når valg av datamodell er gjort, må sentrale spørsmål besvares. Det skal sikre at modellens design ivaretar egenskapene ved et prosjekts livssyklus hos HP Services.

#### 6.2.1 Prosjekt og Faktureringsrutiner

Størrelse, faktureringsrutiner og antall parallelle prosjekter vil avgjøre designet av databasemodell. Spørsmålene besvares og utdypes i detalj. Svarene HP Services ga (i kursiv) vil avgjøre designet av databasemodellen.

- Hva er gjennomsnittlig prosjektstørrelse sett i forhold til antall konsulenter per prosjekt?
  - *Det vil sannsynligvis være flest prosjekter med kun en konsulent, men det må være muligheter for at flere konsulenter kan registrere timer på samme prosjekt (jobbe på flere prosjekt parallelt og flere konsulenter på samme prosjekt).*

Det må i databasemodellen bli tatt høyde for at det i enkelte tilfeller kan være aktuelt med flere konsulenter per prosjekt, og at en konsulent kan jobbe på flere prosjekt. I datamodellen modelleres det som vist på Figur 13.



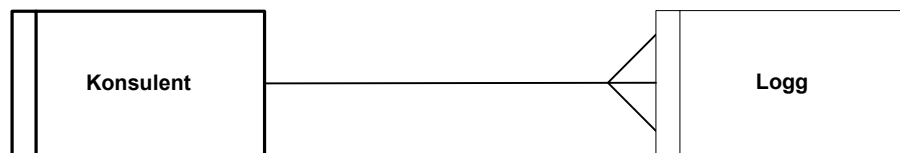
Figur 13: *Prosjekt-Konsulentmodell*

Ser av Prosjekt-Konsulentmodell at konsulent kan jobbe på et eller flere parallelle prosjekt, og at et prosjekt kan ha en eller flere konsulenter. Når modellen tillater at konsulenter jobber på flere prosjekt samtidig, øker skalerbarheten og adaptiviteten til

systemet. Slik øker tilpasningsdyktigheten til andre prosjektorganisasjoner. På den måten skalerer systemet godt på forandringer når det gjelder antall konsulenter per prosjekt og hvordan dette administreres ved gjennomføring av prosjekt. Det vil jo være aktuelt å sette flere konsulenter på et prosjekt ved hektiske faser.

- Kan konsulentene jobbe på flere prosjekt, og av den grunn måtte fakturere flere prosjekt parallelt? Hvis, fører konsulenten en logg gjennom måneden, slik at fakturerte timer knyttes opp mot prosjekter i loggen eller brukes en logg per prosjekt (flere logger)?
  - *Flere prosjekter parallelt er veldig vanlig. Det skal da føres en logg per prosjekt.*

Det i datamodell tas hensyn til at det i enkelte tilfeller er aktuelt med flere logger per konsulent. I databasemodellen modelleres det som vist på Figur 14.



**Figur 14: Konsulent-Loggmodell**

Ser av Konsulent-Loggmodell at en konsulent kan føre timer på flere logger. Det gjør at modellen kan håndtere at konsulenter kan jobbe på flere prosjekt parallelt. Uten denne muligheten ville derfor systemet ikke vært særlig skalerbart, med tanke på at konsulenter skal kunne fakturere på flere parallelle prosjekt.

- Fakturerer flere konsulenter mot samme prosjekt? Hvis, fører de felles logg med ansattnummer som referanse opp mot fakturerte timer, eller fører hver konsulent sin logg?
  - *Her kan det være begge deler, både felles logg og en logg per konsulent selv om de jobber i samme hovedprosjekt.*

Her må det i databasemodellen bli tatt hensyn til at det i enkelte tilfeller kan være en eller flere konsulenter per logg. I datamodellen modelleres det som vist på Figur 15.



**Figur 15: Logg-Konsulentmodell**

Ser av Logg-Konsulentmodell at en logg tillater at en eller flere konsulenter fakturerer på samme logg. Det åpner muligheten for at prosjekter kan gjennomføres med en felles logg for alle konsulenter på samme prosjekt, eller at hver konsulent fører sin egen logg. Systemet gir derfor prosjektleder og kunde mulighet å velge mellom to faktureringsmodeller. Det gjør systemet mer fleksibelt i forbindelse med fakturering. Systemet kan på den måten tilpasses ulike organisasjoners ønsker og krav, og er dermed mer fleksibelt og tilpasningsdyktig.

- Skal konsulenten ha ansvar for å levere logg ved månedsavslutning? Hva gjøres om dette ikke følges opp? Faktureres timene på neste måneds logg?
  - *Hvis ikke annet er avtalt skal logger ajourføres og avsluttes på månedsbasis. Manglende oppfølging fra konsulentens side er i såfall en personalmessig utfordring for nærmeste leder. Timene som av en eller annen grunn ikke er tatt med, skal føres på neste måneds logg.*

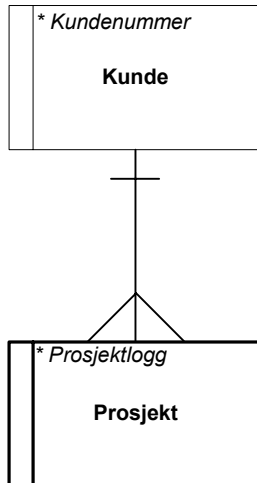
Påvirker ikke datamodellen.

- Hvordan ser man for seg at det skal sikres at timer blir ført på riktig logg? Slik at man ikke må etterfakturere timer måneden etter, eller glemmer man i så fall disse timene?
  - *Konsulenten har ansvar for å føre sine egne logger på riktig måte, på riktig logg og til riktig tid. I prosjekter med prosjektleder vil prosjektlederen ha et kontrollansvar.*

Systemet skal ikke gi føringer for konsulenten på loggføring. Dette påvirker derfor ikke design av databasemodellen.

### 6.2.2 Prosjekt-Kundemodell

Før databasemodellene settes sammen til en total databaseenhet, må et forhold mellom kunde og prosjekt bestemmes. Se Figur 16.



**Figur 16: Prosjekt-Kundemodell**

Ser av Prosjekt-kundemodell at en kunde kan ha flere parallelle prosjekt, men at et prosjekt bare kan ha en kunde. Denne begrensningen representerer en svakhet. Begrensningen er satt for å gjøre designet enklere. Det er dessuten den mest vanlige relasjonen mellom prosjekt og kunde. En slik svakhet vil la seg løse med litt fantasi når det registreres to kunder som er sammen om et prosjekt.

### 6.2.3 Mål

Når det gjelder målene for systemet var mange allerede kjent. Det var likevel nødvendig å få utdypet noen mål.

Mål allerede kjent:

- Produsere dokumenter som manuelt brukes for å registrere faktura i disse systemene
- Fleksibel og enkelt, det må være enkelt for konsulentene å legge inn data, de må kunne gjøre det hjemmefra, fra kunde ol.
- Må være robust - Det må ikke være fare for tap av data
- Må være enkel å forandre i forhold til endrete behov til funksjonalitet og økning i brukere
- Skal bygge på hylleware teknologi
- Det må være enkelt å legge inn funksjonalitet i forhold til beslutningsstøtte data.
- Det må være enkelt å integrere med maler i word/Excel for oversendelse til fakturering og til kunde for godkjennelse

Selv om mye av systemets mål og intensjoner var kjent eksisterte det noen spørsmål. Av den grunn ble følgende spørsmål besvart og utdypet i detalj.

- Er det spesielle krav til faktureringsgrad?

- *Hvis du mener faktureringsgrad per konsulent er ikke det ikke satt krav til dette ennå. Muligens etterhvert, men det trenger du ikke ta hensyn til nå.*

Påvirker ikke design av databasemodell i denne omgang.

- Hva skal systemet gi? (evt. strategiske-, effektivitets- eller produktivitets forbedringer)
  - *Se under "mål allerede kjent". I tillegg skal det være muligheter for rapportering*

Data nødvendig for denne tjenesten ivaretas av entiteter, som allerede er i databasemodellen. Rapporteringsfunksjonalitet påvirker derfor ikke modellen for databasen. Rapporteringsdata hentes ut i forbindelse med generering av rapport.

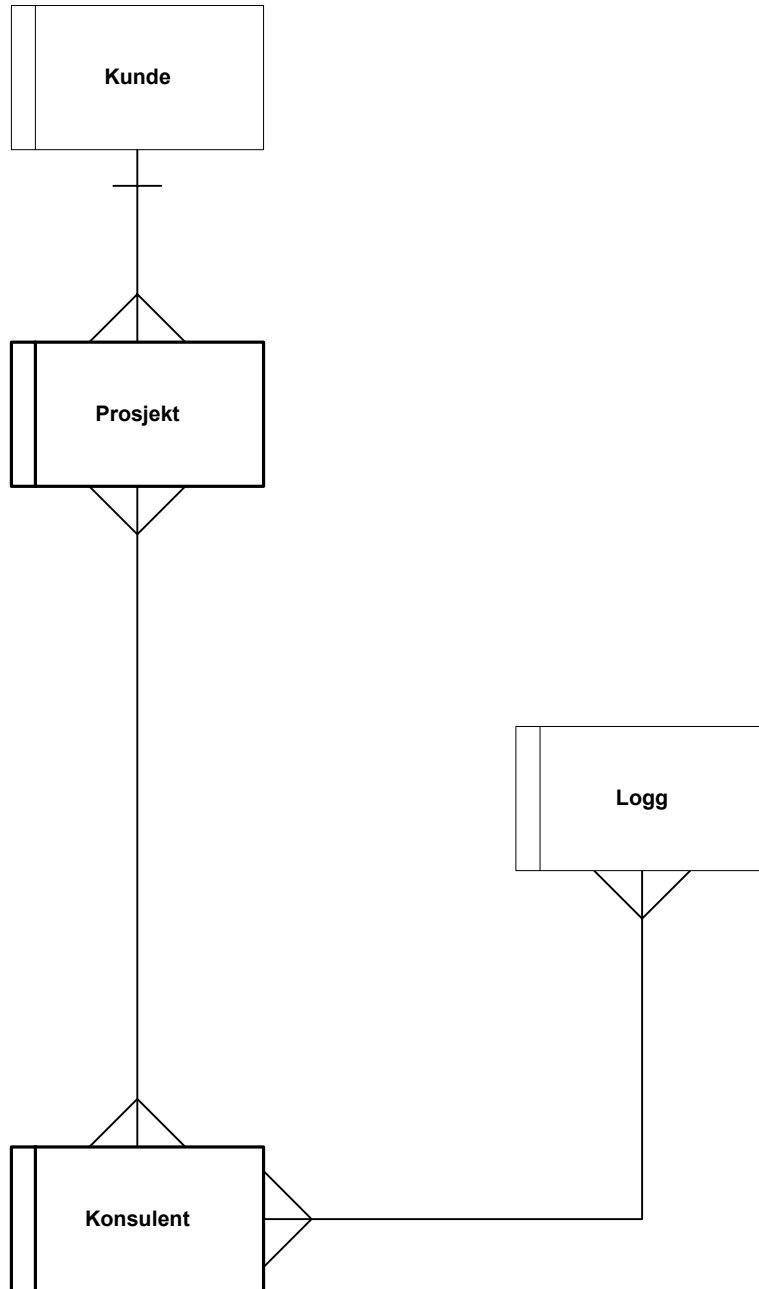
- Skal det være mulig å se differansen på fakturerte timer mot avtalt totalt fakturerbare timer med kunde?
  - *Dette vil vi nok kunne ta ut i en rapport. I utgangspunktet er det ikke tenkt at avtalte totale timer skal registreres (bortsett fra som ren tekst under beskrivelsen av prosjektet, kanskje.)*

Når totalt antall timer fakturert på et prosjekt ikke kan sees under faktureringsprosessen, vil konsulenten få redusert mulighet til å informere overordnede om manglende tid til å gjennomføre prosjektet. Det vil redusere mulighetene for beslutningsstøtte i HPFaktura. Prototyp tilfredsstillter derfor ikke kravet til beslutningsstøtte.

### **6.3 Endelig databasemodell**

Settes alle modellene sammen til en total databasemodell vil modellen bli som vist i Figur 17.





Figur 17: Databasemodell for systemet

### 6.4 Forutsetninger i forbindelse med design

I prototyp ligger ingen begrensninger, men stor grad av frihet. Det betyr at konsulenten fritt kan opprette en logg når prosjektet er registrert. Når loggen er opprettet kan transaksjoner føres i loggen. Det gis full frihet til å føre transaksjoner i åpne logger, slik at konsulenter kan føre transaksjoner i logger opprettet av andre. Systemet tilbyr ingen automatikk i lukking av loggene. Det medfører at det er mulig å registrere transaksjoner på logger så lenge de er åpne. Når en logg stenges vil transaksjoner ikke kunne registreres i

loggen. Skal det faktureres mer på prosjektet må en ny logg opprettes. Det er ingen mulighet til å åpne logger som er lukket for etterregistrering og rettelser.

### **6.5 Databasen i et systemperspektiv**

Designet av databasemodellen bygger på HP Services' valg av datamodell, og svar gitt på sentrale spørsmål knyttet til design av databasemodellen. Spørsmålene sikrer at modellen har de nødvendige kvaliteter. Kvalitetene til modellen sikrer at ulike prosjektorganiseringer knyttet til gjennomføring av et prosjekt ivaretas. Store og små prosjekt medfører gjerne forskjellig gjennomføring, og det er derfor viktig at modellen kan ivareta ulike prosjektorganiseringer.

HP Services valgte modellen som har til intensjon å fokusere på prosjekt og konsulent, som er typiske entiteter i en konsulentorganisasjon. Databasemodellen designes derfor rundt entitetene prosjekt og konsulent. Det ses tydelig i modellen ved at alle entiteter assosieres mot prosjekt eller konsulent. Intensjonen med modellen er at prosjekt og konsulent skal bli satt i et nytt perspektiv. Modellen ønsker ikke å fokusere på fakturerbare timer, men på enhetene som til syvende og sist utgjør inntektene, prosjekt og konsulent. Ved å sette prosjekt og konsulent i sentrum er intensjonen at det skal gjenspeiles i systemet. Effekten av en slik fokusdreining kan være at trivselsfaktoren øker for konsulent, og at kunden som er knyttet til prosjektet opplever en mer engasjert konsulent. Når trivselsfaktoren og engasjementet hos konsulenten er godt kan synergieffekten være økt fakturering. Det kan føre til høyere faktureringsgrad og godt fornøyde kunder.

På den andre siden kan en forandring av faktureringsfokuset mot mer sosiotekniske elementer i prosjektorganiseringen få andre konsekvenser. Redusert press på fakturering kan gjøre prosjektarbeid til en "hvilestol". "Hvilestolen" kan gi en markant svikt i inntektsgrunnlaget til konsulentorganisasjonen. Da er det viktig å ha et belønningssystem i organisasjonen som hindrer at "hvilestolen" blir et faktum. Mulige tiltak mot "hvilestolen" er en bonusordning etter antall fakturerte timer. En slik bonusordning egner seg ikke for alle organisasjoner. Problemer kan oppstå i tilknytning til stillinger som inngår som en del av støttefunksjonene i organisasjonen. Løsningen kan være ulike bonusordninger innad i konsulentorganisasjonen.

Forskjellige bonusordninger kan gjøre at det for enkelte føles som "gresset er grønnere på andre siden av gjerdet". Da kan bonusordningene grave sin egen grav, og man står tilbake med et system som ikke fungerer. Tiltak mot "hvilestolen" må derfor løses innad i hver enkelt organisasjon, gjerne i samråd med delene av organisasjonen som rammes, og slik dras de ansatte med i prosessen. Utviklingen foregår da etter TQM ledelsesprinsippet. Problemet når de ansatte får stor medbestemmelse er at ledelsen føler at de mister sin maktposisjon.

Adaptiviteten til systemet kan ses på i to sammenhenger, i *sanntid* og på *designnivå*. Disse behandles hver for seg, slik at diskusjonen blir oversiktlig.

### 6.5.1 *Adaptivitet i sanntid*

Sanntid handler om systemets tilpasningsdyktighet under bruk. Prosjekt- organisering og gjennomføring vil avhenge av flere faktorer. *Størrelse, tid, teknologi, kompleksitet, kunde, omgivelser* er eksempler på slike faktorer. Det er faktorer som påvirker et prosjekt og oppdragsgiver. Systemet må til enhver tid tilpasses de ulike faktorene. Tilpasningsevnen øker adaptiviteten, og er helt nødvendig for å oppnå et fungerende system.

Faktorene påvirker hvor mange konsulenter som jobber på et prosjekt. Derfor tar datamodellen høyde for at antall konsulenter på et prosjekt kan variere, også under gjennomføring.

Kunden bør ha mulighet til å påvirke hvordan fakturering skal gjennomføres. Derfor gir systemets datamodell mulighet for at konsulenter kan fakturere timer på felles logg eller at hver konsulent har sin logg.

Faktorene påvirker prosjektets levetid, og levetiden kan ofte være vanskelig å forutsi. Systemet må kunne tilpasses forandringer i prosjekts levetid. Det betyr at det må være mulig å fakturere timer etter at prosjektet skulle vært ferdigstilt. Det ligger derfor ingen automatikk i systemet som lukker et prosjekt ved forventet avslutningsdato. Dette løses ved at prosjektenes status settes manuelt. På den måten vil det være mulig å fakturere timer på et prosjekt så lenge kunden tillater.

### 6.5.2 *Adaptivitet på designnivå*

Designnivå handler om å gi systemet forutsetninger for å være adaptivt ved hjelp av teknologi. For å gjøre systemet tilpasningsdyktig for prosjektorganisasjoner som varierer i størrelse må systemet være tilpasningsdyktig til ulike organisasjoners behov. Behovene kan deles opp i tre sentrale nøkkelord, *Skalerbarhet, forretningskritiske krav og prototyping*.

Faktorene påvirker valg av database. Tabell 1 presenterer noen egenskaper som er verdt å ta hensyn til når du designer databaser i Ms Access. Den gir god veiledning i hvilken databasemotor som bør velges med hensyn på systemkrav. Disse kravene ble benyttet for å gjøre et valg av database for systemet.

**Tabell 1: Jet Data Engine versus Microsoft Data Engine (MSDE)**

<b>Requirement</b>	<b>SQL Server (use MSDE if these are future requirements)</b>	<b>Microsoft Access (Jet)</b>
<b>Scalability</b>	<ul style="list-style-type: none"> <li>• SMP Support</li> <li>• Virtually unlimited number of concurrent users</li> <li>• Terabyte levels of data</li> <li>• Transaction Logging</li> </ul>	<ul style="list-style-type: none"> <li>• No SMP support</li> <li>• Maximum of 255 users</li> <li>• 2 GB of data</li> <li>• No transaction logging</li> </ul>
<b>Business Critical</b>	<ul style="list-style-type: none"> <li>• 7X24 support and QFE</li> <li>• Point-in-time recovery</li> <li>• Guaranteed transaction integrity</li> <li>• Built-in fault tolerance</li> <li>• Security integrated with Windows NT</li> </ul>	<ul style="list-style-type: none"> <li>• No 7X24 support or QFE</li> <li>• Recoverable to last backup</li> <li>• No transaction logging</li> <li>• No integrated security with Windows NT</li> </ul>
<b>Rapid Application Prototyping</b>	Access is UI for both engines and offers WYSIWIG database tools and built-in forms generation.	

*Skalerbarhet*

Skalerbarheten til et system ivaretas av egenskapene til databasen. Databasen som er valgt er Microsoft Data Engine (MSDE). MSDE databasen er 100% kompatibel med Ms SQL Server 7.0, og har derfor indirekte lik ytelse som en Ms SQL Server database.

Databasemotoren er svært adaptiv på flere områder. Den er egnet som database for system som kan inngå i en fremtidig systemutvidelse. Kompatibiliteten med Ms SQL Server gjør at systemet enklere integreres med eksisterende system, for eksempel fakturering-

og lønningssystem, hvis systemene ligger på en Ms SQL Server kompatibel database.

MSDE mangler støtte for Symmetric Multiprocessing (SMP). Ved fremtidig behov for SMP kan kompatibiliteten til Ms SQL Server benyttes. SMP innebærer å oppgradere server med flere prosessorer, minne, disker og nettverkskort.

MSDE kan betjene inntil fem parallelle brukere. Det gir systemet begrensede muligheter til å vokse i antall parallelle brukere. Mulighetene for en MSDE database til å skalere antall brukere er til stede. Det ivaretas av kompatibiliteten med Ms SQL Server. Det gir systemet virtuelt ubegrenset antall parallelle brukere.

Når antall brukere øker vil datamengden i databasen øke. Det åpner for at databasen må kunne behandle store mengder data. MSDE kan behandle datamengder opp til 2G. For store databasesystem med mange brukere blir det for lite kapasitet. Igjen gir kompatibiliteten med Ms SQL Server, MSDE databasen mulighet for å skalere datamengden opp til Terabytes av data.

Når antall parallelle brukere blir større, øker faren for transaksjonsfeil. MSDE har ingen transaksjonslogging for bruk ved datagjenvinning. Støtte for transaksjonslogging oppnås ved å oppgradere MSDE databasen til Ms SQL Server. Det vil gi databasen transaksjonslogging. Ved transaksjonslogging kan data overføres til databasen uten fare for tap av data. Det gjør databasen mer pålitelig, noe som absolutt er nødvendig for databaser som tjener mange parallelle brukere. Når transaksjonsfeil oppstår foretar databasen en "point in time" datagjenvinning. Slik vil en datagjenvinning foregå uten at brukeren merker noe.

### *Forretningskritiske krav*

Forretningskritiske krav ved en database ivaretas av en kombinasjon mellom skalerbarhet og andre støttetjenester bundet til databasetypen. En forretningskritisk database bør ha 7x24 support. Det finnes ikke for MSDE databasen. En oppgradering til Ms SQL Server vil tilby denne tjenesten. 7x24 support kan løse problemer som måtte oppstå raskt og effektivt. På den måten reduseres de mest forretningskritiske skadene ved en eventuell databasekrasj.

Som en del av denne tjenesten støtter Ms SQL Server Quick Fix Engineering (QFE). QFE gjør det hurtig og enkelt å kjøre bugfix på databasen. Slik kan de mest forretningskritiske feilene ved databasen hurtig oppdateres eller repareres.

Transaksjonsintegritet garanteres ikke ved MSDE. Kreves det integritet av datatransaksjonene må Ms SQL Server benyttes. Ved den databasen kan bedriftsfortrolige informasjon skrives til databasen uten fare for tap av informasjon, og at integriteten forringes.

MSDE databasen har ikke innebygd feiltoleranse. Innebygd feiltoleranse tillater at databasen kan være operativ selv om en server

feiler eller blir tatt ut av tjeneste for vedlikehold. Det sikrer høy oppetid for systemet. Er denne egenskapen ønskelig må Ms SQL server benyttes.

### *Prototyping*

Prototyping for en MSDE database foregår hurtig og effektivt med Ms Access. Ms Access tilbyr brukergrensesnitt for design av en MSDE database. Det gir brukeren et utviklingsmiljø med WYSIWYG (What You See Is What You Get).

Når databasen er ferdig designet, og i bruk, finnes bare en GUI applikasjon for å administrere databasen. Administrasjonen foregår med Ms SQL Server Service Manager. Den gir begrensende administrasjonsmuligheter. Det kan være en begrensende funksjonalitet for en database som vokser seg stor, med mange brukere. Nye behov for administrasjon kan oppstå som følge av at databasen vokser.

Ulike GUI applikasjoner for administrasjon tilbyr ofte ulik funksjonalitet for ulike behov. Kompatibiliteten med Ms SQL Server kan gjøre tilpasningen til endrede behov for funksjonalitet i forbindelse med administrasjon enklere. Ms SQL Server støttes av mange "Service Managere" som tilbyr ulik funksjonalitet for administrasjon.

Ofte kommer endrede behov for ny funksjonalitet som en følge av at systemet vokser. Endrede krav til funksjonalitet vil på den måten ofte føre til at databasen må oppgraders før bristepunktet på ytelse er oppnådd. På den måten unngår en at databasen bryter sammen på grunn av overbelastning.

MSDE har i tillegg begrensede muligheter for fjernadministrasjon. Fjernadministrasjon kan bli nødvendig etter som databasen vokser og nye krav til vedlikehold oppstår. Begrensende mulighet for fjernadministrasjon kan skape problemer i forbindelse med vedlikehold og support. Da vil en systemoppgradering til Ms SQL Server løse problemene. Den har full støtte for fjernadministrasjon og multi server administrasjon.

Referanse [19]

### **7 Innføring, prototyp og plattformens egnethet for adaptiv prosessforbedring**

Ms Outlook er et program som brukes av mange hver dag. Når applikasjonen er lett tilgjengelig fra Outlook kan det være en faktor som er med på å opprettholde kontinuerlig fakturering.

At applikasjonen er tilgjengelig fra Outlook, som er godt etablert blant brukerne, kan gjøre innføringen lettere. Applikasjonen kan virke mindre skremmende når den inngår som en del av et eksisterende system. Frittstående applikasjoner kan på enkelte brukere virke mer skremmende, mens nye funksjoner som kommer i en applikasjon som er godt innarbeidet har lavere terskel for å bli akseptert.

Når det er sagt, er applikasjonen tenkt brukt til timeregistrering for IKT konsulenter. Brukerne vil derfor være mer mottakelig og tilpasningsdyktig til nye datatekniske løsninger for støtte i arbeidsprosesser. Derfor kan denne problematikken være ubegrunnet i dette virksomhetsmiljøet. Men innføringsproblematikk på andre nivå kan oppstå. Derfor har jeg foreslått at utviklingen foregår etter TQM som ledelsesprofil.

#### **7.1 Prototypens egnethet for adaptiv prosessforbedring**

I beskrivelsene av prototypen lå det sterke føringer fra oppdragsgiver om skjermbildenes design. Prototypens skjermbilder presenteres i følgende figurer.

Figur 18 viser de ulike skjermbildene i HPFaktura

- Skjermbilde for registrering av kunde
- Skjermbilde for å *opprette prosjekt/logg*
- Skjermbilde for *håndtering av logger*
- Skjermbilde for å *velge prosjekt for timeregistrering*
- Skjermbilde for registrering av timer

Systemet trenger ulike input parametere for å utføre faktureringen. Det er ivaretatt i prototyp. For at systemet skal være egnet for adaptiv prosessforbedring må det være tilpasningsdyktig til endrede input parametere og endrede administrative rutiner rundt faktureringen. Det påvirker input parametere og programkoden som beregner faktura.

Parametere som er av betydning for hvordan faktureringen foregår velges fra lister. Listene er dynamiske. Løsningen med dynamiske lister gjør at systemet raskt kan konfigureres til å bruke andre parametere i faktureringen. Slik er systemet adaptivt til endring i input parameterne. Endring av listene er en del av endrede rutiner for beregning av faktureringen. Teknisk løses det ved at valgene i listene er generert fra tabeller i databasen. Det betyr at de kan endres uten at utvikleren må endre applikasjonskoden. På den måten unngås omfattende oppdateringer på konsulentenes datamaskiner ved endring av input parametere. Det gjør at oppdateringen blir gjeldende for alle

## **Innføring, prototyp og plattformens egnethet for adaptiv prosessforbedring**

---

maskiner samtidig. Endringsprosessen blir på den måten mer fleksibel. Dessuten gjør det prosessen mindre ressurskrevende.

Hver liste har sin egen tabell. Det kunne vært løst med en tabell, men for å bevare oversikten i koden ble det valgt en løsning med en tabell per liste. Det gjør applikasjonskoden mer lettlest.

Koden som beregner faktureringsgrunnlaget, med hensyn på input parametere, må oppdateres når listene endres. Koden som beregner faktureringsgrunnlaget bør ligge på databasesiden. Det gjør at oppdateringen blir gjeldende for alle maskiner samtidig. Endringsprosessen blir på den måten mer fleksibel. Dessuten gjør det prosessen mindre ressurskrevende. Koden plasseres i SQL funksjoner i databasen. Når beregning skal utføres kalles gjeldende funksjon med tilhørende SQL setning.

Når alle input parametere og beregningsfunksjoner ligger på databasesiden oppnås en oversiktlig struktur for systemendringene. Årsaken er at forandringene foretas på et felles sted.

Brukergrensesnittet vil ikke påvirkes. Det er kun innholdet i listene og SQL funksjonene som endres.

I systemet ligger det ingen begrensninger på når timeregistrering skal utføres. I prototyp er det lagt opp til frihet under ansvar. Det er opp til hver enkelt konsulent å følge gjeldene rutiner for fakturering. På den måten er systemet svært tilpasningsdyktig til endringer rundt det administrative ved timeregistrering.

Likevel er systemet lite egnet til prosessforbedring. Systemet har et relativt ukjent brukergrensesnitt med tanke på at det skal brukes til timeregistrering. Det kan redusere brukeraksepten. Det mest nærliggende er å benytte gridlayout for å designe registrerings skjema. Ved bruk av gridlayout ville systemet hatt større likhetstrekk med dagens manuelle system, og da ville mulighetene for brukeraksept steget.

I dag er systemet uoversiktlig. Registrering av timer er tungvint, og innebærer flere skjermbilder. Det er ikke mulig å korrigere registrerte timer. Ønskes tilbakemelding på hva som er registrert må det genereres rapport.

Ved bruk av gridlayout vil registreringen bli oversiktlig. Gridlayout forenkler presentasjon av registrerte timer. Det gjør korrigering av timer enklere. Presentasjon av registrerte timer i gridlayout tilfredsstillende behovene for tilbakemelding.

Systemet gir ingen mulighet for beslutningsstøtte run time. Årsaken er at systemet ikke gir tilbakemelding på totalt avtalt fakturerbare timer med kunde, tidligere registrerte timer og differanse mellom fakturerte timer og totalt avtalt fakturerbare timer.



## Innføring, prototyp og plattformens egnethet for adaptiv prosessforbedring

På den måten kan data for beslutningsstøtte bli forsinket. Beslutningsdata må tas ut av andre sammenhenger og antagelser, for eksempel ut fra registrerte timer.

HP Services ble spurt om de ønsket mulighet for beslutningsstøtte run time. De mente det var tilfredsstillende å få disse opplysningene ut i en rapport (se kap. 6.2.3).

HPFaktura 1.0 - Registrering kunde

Kundeopplysninger

Kunde informasjon

Kundennummer Kundernavn Telefonnummer

Fakturaadresse

Gateadresse Postnummer Sted

Besøksadresse

Gateadresse Postnummer Sted

Referansepersoner

Referanseperson E-post Telefon

Registrer Cancel

Skjerm bilde for registrering av kunde

HPFaktura 1.0 - Registrering kunde

Logg informasjon

March 2001 March 2001

Mon	Tue	Wed	Thu	Fri	Sat	Sun
25	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Velg kundennummer

Generer Logg ID

Logg ID og Status

Logg ID

Prosjekt Status

Skjerm bilde for å opprette prosjekt/logg

HPFaktura 1.0 - Registrering kunde

Avsluttende logg informasjon

March 2001 March 2001

Mon	Tue	Wed	Thu	Fri	Sat	Sun
25	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Velg Konsulent

Velg Logg ID

Prosjekt Status

Endre Status

Avslutt Logg Vis Logg Delt Logg Endre Status

Skjerm bilde for håndtering av logger

HPFaktura 1.0 - Registrering kunde

Velg Logg ID

Prosjekt - tittel/beskrivelse

Prosjekt Status

Kunde informasjon

Kundennummer Kundernavn Telefonnummer

Fakturaadresse

Gateadresse Postnummer Sted

Besøksadresse

Gateadresse Postnummer Sted

Referansepersoner

Referanseperson Referanseperson

Skjerm bilde for å velge prosjekt for timeregistrering

HPFaktura 1.0 - Registrering aktivitet

Registrering aktivitet

March 2001 March 2001

Mon	Tue	Wed	Thu	Fri	Sat	Sun
25	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Navn konsulent

Ang Task Id Beskrivelse aktivitet

RI (Revenue)  
TR (Training)  
CO (Cost)  
TA (Travel)

Kalkuler aktiviteter og reise

Skjerm bilde for registrering av timer

HPFaktura 1.0 - Registrering aktivitet

Registrering aktivitet

March 2001 March 2001

Mon	Tue	Wed	Thu	Fri	Sat	Sun
25	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Navn konsulent

Ang Task Id Beskrivelse aktivitet

RI (Revenue)  
TR (Training)  
CO (Cost)  
TA (Travel)

Kalkuler aktiviteter og reise

Antall Timer Timesats Sum Totalsum

Skjerm bilde for registrering av timer med "task id" valgt

Figur 18: De ulike skjerm bildene i HPFaktura

### **7.2 Plattformens egnethet for adaptiv prosessforbedring**

Utviklingsplattformen kan brukes til utvikling av et adaptivt WfMS for bruk i prosessforbedring, men har begrensede muligheter. Slik systemet fremstår i dag, har ikke begrensede muligheter vært en hindring for ønsket funksjonalitet.

Når det gjelder faktureringsystemer er gridlayout å foretrekke. Plattformen har mulighet for utvikling med ActiveX komponenter. Det gjør det mulig å benytte gridlayout. Under uttesting med gridlayout, i forbindelse med et forslag til oppdragsgiver, ble det oppdaget mangler som kompliserte programmeringen. Dette forslaget ble avvist, så manglene fikk ikke konsekvenser for nåværende prototyp.

Blant annet mangler "onFocus-event". Det brukes for å legge skjult funksjonalitet i cellene. Skjult funksjonalitet i celler, kommer til syne når cellen settes i fokus. Hvis prototyp fortsatt skal utvikles over "samme lest", må det tenkes nøye gjennom fremtiden for systemet. Er det ønskelig å legge inn mer avansert funksjonalitet anbefales en annen plattform. Det er rimelig å tro at fremtiden kan bringe ønsker om mer avansert funksjonalitet. Nåværende plattform er da ikke tilstrekkelig.

### **8 HPFaktura – Evaluering og diskusjon**

Et hvert utviklingsprosjekt på prototypnivå vil resultere i nye ideer og oppdagelse av svakheter. Derfor er det nødvendig å presentere nye ideer og forslag til forbedringer i en utviklingsrapport.

Nåværende prototyp er uoversiktlig og har store mangler. Slik applikasjonen fremstår i dag minner den for lite om tradisjonelle skjema for registrering av timer. Det rom for store forbedringer som gjør den bedre egnet som et verktøy for adaptiv prosessforbedring i en fakturerings situasjon.

#### **8.1 Feedback**

Nåværende versjon gir ingen feedback på hva som er registrert. I tradisjonelle registrerings skjema blir timeregistreringer presentert uke- for ukedag. Da kan en hel ukes registreringer vises. På den måten vil tradisjonelle skjema kvalitetssikre faktureringen. Kvalitetssikringen innebærer at konsulenten lett ser om det ble fakturert timer sist han var tilstede på jobb, og om det er registrert riktig antall.

Videre foreslår jeg at det skal være mulig å traversere uke- for ukenummer tilbake på et prosjekt. På den måten kan hele din siste måneds fakturering raskt hentes frem, uke- for ukenummer.

Det vil være et behov for presentasjon av en hel måneds registreringer på skjerm. På den måten kan du traversere gjennom hele din faktureringsprofil på et prosjekt. Da blir det enkelt å hente frem historiske data. Det gjør det enklere å få et helhetsinntrykk av hva som er fakturert på et prosjekt. På den måten får konsulenten bedre oversikt.

Videre må det være enkelt å korrigere registrerte timer. Feilretting av allerede registrerte timer foregår også i skjermbildet for registrering. Da kan presentasjon, registrering og korrigerings av timer på en fornuftig måte foretas i samme skjermbilde.

Denne funksjonaliteten oppnås best når registrering og presentasjon foregår i gridlayout.

#### **8.2 Beslutningsstøtte**

Videre foreslår jeg prototyp forbedret på støtte rundt den *formelle koordineringen* og rundt de *økonomiske transaksjonene*. Disse støtteformene vil utgjøre beslutningsstøtten som systemet tilbyr. Beslutningsstøtten som tilbys vil genereres på basis av transaksjonsdataene registrert på et prosjekt og opplysninger konsulent registrer. Dataene vil hentes inn, og prosesseres slik at beslutningsstøtte kan tilbys. Målene med beslutningsstøtten er [11]:

## HPFaktura – Evaluering og diskusjon

- Oppnå en skalerbar prosess fra små uformelle prosjekt til relativt store tverrfaglige prosjekt med mange konsulenter fra ulike avdelinger eller partnerskap.
- Tidlig fokus på korrigerende og konfliktløsende aktiviteter.
- Forbedre kvalitet og ytelse på faktureringen.
- Automatisering og standardisering av transaksjoner mellom avdelinger og partnerskap.

Sentralt i modellen som beslutningsstøtten baseres på er *aktive roller*, *transaksjonsstatus* og *forespørsler* [11].

Aktive roller:

**Prosjekt Konsulent (PK):** Registrerer timer så nøyaktig som mulig på et prosjekt. Regelmessig estimerer PK gjenstående tidsforbruk for å ferdigstille oppgaven.

**Prosjekt Leder (PL):** Følger opp PK og initierer korrigerende aktiviteter mot kunder og PK. Rapporterer til PD når assistanse er nødvendig.

**Prosjekt Koordinator (PKO):** Følger opp og assisterer PL og PK. Assisterer i konflikter og korrigerer av feil. Rapporterer til PD.

**Prosjekt Direktør (PD):** Godkjenner forespørsler. Assisterer PL og PK med uløste konflikter og problemer. Assisterer PL med korrigerende aktiviteter mot kunde, som for eksempel reforhandlinger.

Transaksjonsstatus:

Registrerte timer og kostnader er overført til følgende koder for status.

**Tabell 2: De ulike transaksjonsstatusene hos et prosjekt**

<b>NAW</b> <b>NAR</b>	Not Accepted, Waiting Not Accepted, Resolved	N hours, no debit
<b>ANI</b> <b>AW</b> <b>AI</b>	Accepted, Not to be Invoiced Accepted, Waiting Accepted, ready for Invoice	A hours, debit to project
<b>I</b>	Invoiced	I hours, credit to project

Foresørsler:

**Tabell 3: De ulike forespørselene på et prosjekt**

<b>FK</b>	Forpliktelser mot kunde (kontrakt)
<b>FPL</b>	Totalt fakturerbare timer på et prosjekt. Foreslått av <u>PL</u> (FPL) som retningslinje for PK (debit på prosjekt)
<b>FPK</b>	Estimert antall gjenstående timer for å ferdigstille en oppgave. Foreslått av <u>PK</u> (FPK)

Prosjektlederen (PL) administrerer og kontrollerer FK og FPL. Ved å ha denne kontrollen har prosjektlederen indirekte kontroll over statusen og registrerte timer på et prosjekt.

Timer innenfor rammene av FK feltet er automatisk godkjent. Timene i FK er kontraktsfestet. Timene mellom FK og FPL er godkjent brukt på prosjektet, men skal ikke faktureres kunde. De må gå på firmaets regning.

Prosjektkonsulenten (PK) kontrollerer FPK. Her angir konsulenten et estimert antall timer som er nødvendig for å gjøre ferdig oppdrag. Hvis FPL overstiges, uten i god tid å ha justert FPK, har ikke konsulenten oppfylt sine forpliktelser.

Figur 19 presenterer et forslag til timeregistreringsskjema for PK. Skjemaet skal ivareta den *formelle koordineringen* og gi en tilfredsstillende støtte for automatiserte *økonomiske transaksjoner* i en organisasjon.

Prosjekt	Godtatt		Timer registrert i uke #							Brukt		Gjenstående	
	FK	FPL	Mandag	Tirsdag	Onsdag	Torsdag	Fredag	Lørdag	Søndag	Per.	Total	FPL	FPK
Prosjektcode.navn	350	375	8	5	4	8	6			31	225	150	175
Prosjektcode.navn	250	300			4					4	75	225	
Prosjektcode.navn	150	180		3			2			5	25	155	
Prosjektcode.navn	NA	25				2				2	10	15	
Sum			8	8	8	10	8	0	0	42			

**Figur 19: Timeregistreringsskjema med beslutningsstøtte**

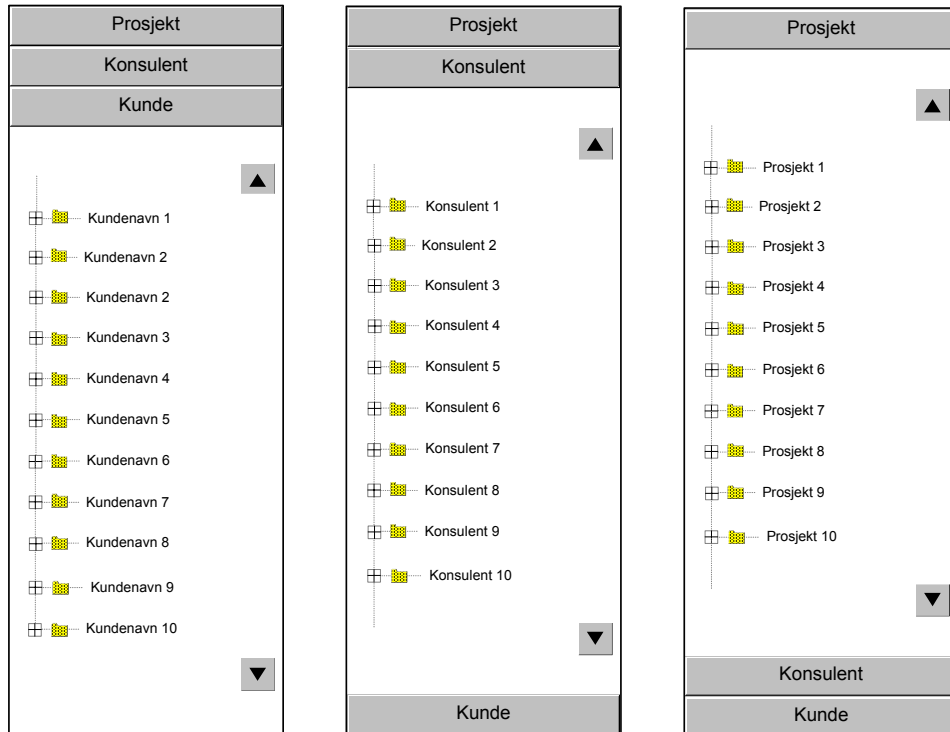
De hvite feltene utgjør registreringsskjemaet for PK. De grå feltene gir tilbakemelding til PK, PL, PKO og PD. Tilbakemeldingene som gis i skjemaet skal gi en bedre formell koordinering og støtte rundt de økonomiske transaksjonene, det vil si beslutningsstøtten.

### 8.3 Identifikasjonsparametere

For å utføre forskjellige oppgaver mot et prosjekt må identifikasjon om kunde, konsulent og prosjekt innhentes. I prototyp foregår det med dynamiske lister i skjermbildet for gjeldende oppgave. Det betyr at samme liste finnes i flere utgaver i forskjellige skjermbilder. Det kan være forvirrende. Derfor foreslår jeg bruk av et helt nytt navigeringssystem.

Navigeringssystemet skal ta hånd om kunde, konsulent- og prosjektidentifikasjon, slik at identifikasjonen hentes på samme sted, og i samme meny hver gang, uavhengig av skjermbilde. Figur 20

presenterer navigasjonssystemet, og de tre modusene som det kan opptre i.



**Figur 20: Navigeringssystem for Kunde-, Konsulent- og Prosjektidentifikasjon**

Navigasjonssystemet er dynamisk. Det fungerer på samme måte som de dynamiske listene. Forskjellen ligger i at navigeringssystemet integrerer de tre unike identifikatorene i hver trenode, uavhengig av om menyen er i kunde-, konsulent- eller prosjektmodus, se figur 20.

Ved utførelse av forskjellige oppgaver skal kunde-, konsulent- og prosjektidentifikasjon hentes fra navigeringssystemet. Hvem av de tre navigasjonsmodusene det tas utgangspunkt i, vil avhenge av hvilken oppgave som skal utføres, og hva som er hensiktsmessig.

For eksempel vil det under arbeid med timeregistrering være hensiktsmessig å ta utgangspunkt i *prosjekt*. Nå vil konsulenten hurtig kunne velge prosjekt å fakturere, samtidig som de unike identifikasjonsparametere er lett tilgjengelige.

I tillegg til unike identifikasjonsparametere kan metainformasjon om *kunde*, *konsulent* og *prosjekt* hentes frem via navigasjonsmenyen. Metainformasjonen kan leses, skrives ut eller editeres.

- Velger du *kunde* kan metainformasjon om en gitt kunde hentes frem i system panel
- Velger du *konsulent* kan metainformasjon om en gitt konsulent hentes frem i system panel
- Velger du *prosjekt* kan metainformasjon om et gitt prosjekt hentes frem i system panel



**Figur 21: System panel før oppgave er valgt**

Figur 21 viser hvor *system panel* posisjonerer seg i applikasjonen. Metainformasjonen for de tre systemelementene, *kunde*, *konsulent* og *prosjekt* vil presenteres der. De ulike oppgavenes skjermbilder presenteres også i system panel.

For å gjøre navigeringen mellom oppgaver oversiktlig er det nødvendig med menyer. De mest brukte oppgaver finnes på "hurtigknapper". Menyene og hurtigknappene skal gjøre systemet mer brukervennlig.

### **8.4 Fraværsmodul**

Hvis det i systemet innføres en fraværsmodul kan den informere andre kolleger om fravær, når du vil være tilbake og en kort informasjon om fraværet. Under registrering av fravær må det være valg for om fraværet er fakturerbart.

Kommunikasjonen mellom fraværsmodul og systemet gjør at dagene du har vært borte fra jobb, som ikke er fakturerbar, automatisk blir inaktive i registreringsskjemaet. Denne funksjonaliteten vil være nyttig under faktureringen og for dine kollegaer.

Funksjonaliteten krever selvfølgelig at den følges opp av brukerne. Følges den ikke opp har den liten verdi. I en større organisasjon vil en slik modul være mer ettertraktet enn i en liten. I mindre organisasjoner er det enklere å ha kontroll på hvor dine kollegaer befinner seg.

## **9 Konklusjon**

En faktureringsprosess påvirkes av interne og eksterne faktorer. *Forutsetninger, ressurstilgang, organisasjon og partnerskap* er elementer som påvirkes av omgivelsene. Et adaptivt WfMS i en faktureringsprosess gjør at prosessen kan være under kontinuerlig forandring og forbedring for å møte endringer i omgivelsene. Det gjør at bedriften til en hver tid har en optimal prosess for fakturering.

Fordelen med implementasjon av WfMS i Ms Outlook er at ny funksjonalitet i godt kjente brukermiljø har større sjanse for brukeraksept. Et nytt selvstendig applikasjon vil ha større problemer med å få den nødvendige brukeraksepten. Det stilles derfor større krav til design og tekniske løsninger.

Prototyp er lite egnet til adaptiv prosessforbedring. Den er uoversiktlig har et uvant brukergrensesnitt. Det ukjente brukergrensesnittet kan redusere brukeraksepten.

Timeregistreringen er tungvint. Brukervennligheten i forbindelse med timeregistrering, tilbakemelding på registrerte timer og korrigerings av registrerte timer må forbedres.

Systemet gir ingen beslutningsstøtte run time. Årsaken er at systemet mangler tilbakemelding på totalt avtalt fakturerbare timer med kunde, tidligere registrerte timer og differanse mellom fakturerte timer og totalt avtalt fakturerbare timer. Når HPFaktura mangler beslutningsstøtte innfris ikke målet med beslutningsstøtte i systemet.

Prototyp må forbedres rundt den *formelle koordineringen* og rundt de *økonomiske transaksjonene*. Det vil danne beslutningsstøtten i systemet.

Bruken av konsulent-, prosjekt- og kundeidentifikator er uoversiktlig. Derfor foreslås et nytt navigeringssystem for å gjøre tilgangen til identifikatorene mer oversiktlig. Navigeringssystemet tilbyr i tillegg mulighet for å hente ut metadata om konsulent, prosjekt og kunde.

Den teknologiske plattformen tilfredsstillende behovene til HPFaktura i nåværende form. Plattformen har svakheter som gjør eventuelle forbedringer vanskelig. VBA har få events til bruk i programmeringen. Det kompliserer implementering av mer avansert funksjonalitet. Manglene vil skape problemer under implementering hvis systemet skal forbedres med gridlayout. Plattformen er uegnet for programmering med gridlayout.

Da valg av database ikke ble Ms Exchange Server 2000 er det ingen logisk knytning mellom Ms Outlook 2000 og HPFaktura. Derfor burde en annen utviklingsplattform vært valgt. Det vil åpne muligheter for mer avansert funksjonalitet i fremtiden om det skulle være behov for det.



**10 Kilder**

- [ Christiansen, Grønland, Methlie. Informasjonsteknologi, -strategi, -styring, -styring. 3. 1 utgave 1999. ISBN 82-456-0266-3  
]
- [ Kåre Hansen, TQM og den norske samarbeidsmodellen- fra lov- og avtalebasert 2 representasjon til konseptbasert medvirkning. RF-1999/356  
]
- [ Rebecca E. Grinter, Workflow Systems: Occasions for success and failure. Bell Labs, Lu 3 Technologies.  
]
- [ Hanne Cecilie Langkaas, Hvordan arbeidsflytsystem støtter arbeidet og ledelsen av 4 arbeidet. Mai 1999.  
]
- [ Levin, Fossen, Gjersvik. Ledelse og teknologi. 1. utgave, 5. opplag, 1999. ISBN 82-00- 5 032973  
]
- [ Knut Håkon Tolleshaug Mørch, Kontekstbasert arbeids- og informasjonsorganisering. M 6 1999.  
]
- [ <http://ikt.hia.no/lars.line/doc/fag/systemeng/systemeng.html>  
7  
]
- [ <http://www.soce.org/>  
8  
]
- [ <http://www.powermfg.com/ipd.html>  
9  
]
- [ Christoph Bussler, A Taxonomy of Adaptive Workflow Management. The University of 1 Georgia, Athens, GA30602-7404  
0  
]
- [ Lars Line, Etablering av distribuert ingeniørarbeid. Teknologiske og organisatoriske 1 erfaringer fra en norsk ingeniørbedrift. ISBN 82-471-0420-2  
1  
]
- [ Levin, Fossen og Gjersvik, Ledelse og Teknologi. 1. Utgave, 5. opplag 1999. ISBN 82-0 1 03297-3  
2  
]
- [ [CSCW - Computer-Supported Cooperative Work](#)  
1  
3  
]
- [ [Avansert Intranett Samarbeide](#)  
1  
4  
]
- [ <http://www.eil.utoronto.ca/tool/BPR.html>  
1  
5  
]

## HPFaktura – Evaluering og diskusjon

[ <http://msdn.microsoft.com/office/>  
1  
6  
]  
[ <http://msdn.microsoft.com/vba/>  
1  
7  
]  
[ <http://msdn.microsoft.com/office/xp/>  
1  
8  
]  
[ <http://msdn.microsoft.com/library/default.asp?URL=/library/backgrnd/html/msdeforvs>  
1  
9  
]  
[ <http://msdn.microsoft.com/library/officedev/off2000/acconAboutMSDE.htm>  
2  
0  
]

## **Vedlegg**

Prototypen viser rammeverket for funksjonalitet. Implementering av funksjonalitet er ikke fullstendig. Videre følger en beskrivelse av implementert funksjonalitet, installasjonsveiledning og kildekode.

### **Implementert funksjonalitet**

For å avslutte HPFaktura må systemclose (X) benyttes.

*Registrering av kunde* er ferdig implementert. Det er ikke lagt funksjonalitet i knappen Cancel.

*Opprett logg* er ferdig implementert.

*Avslutt logg til manuell fakturering* er delvis implementert. Det er ikke lagt funksjonalitet i knappene Avslutt Logg, Vis Logg og Print Logg. Knappen Endre Status er implementert. Statusfeltene i *Opprett logg* og *Avslutt logg til manuell fakturering* vil vise gjeldende status for et prosjekt.

*Registrer fakturagrunnlag* er ikke implementert.

### **Veiledning installasjon**

MSDE [20]:

Installasjonsfilen er lokalisert i \Sql\x86\Setup folder på Ms Office CD  
Kjør filen Setupsql.exe  
Start SQL Server Service Manager  
Dobbelklikk MSSQLServer ikonet på task bar og klikk Start/Continue (Må utføres før kjøring av applikasjon og designarbeid på databasen)

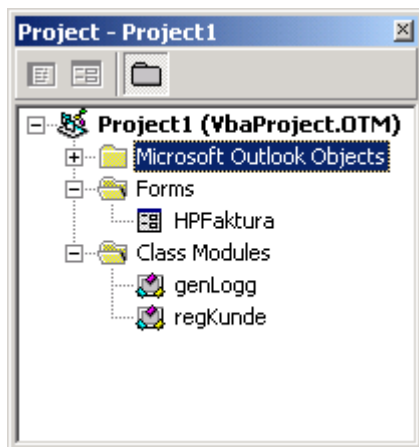
Database:

1. Start ODBC Data Source Administrator
2. Legg til en SQL Server med navn HPFaktura som peker på HP\_faktura.adp

HPFaktura:

1. Start Ms Outlook 2000
2. Trykk ALT + F11
3. Merk katalogen Microsoft Outlook Objekts i Project Explorer. Se Figur 22.
4. Velg File, Import File fra gjeldende lagringssted. Filene som skal importeres er:
  - HPFaktura.frm
  - RegistrerAktivitet.frm

- GenLogg.cls
  - RegKunde.cls
5. Merk HPFaktura i katalogen Forms i Project Explorer
  6. Høyreklikk, og velg View Object
  7. Velg Run, Run Sub/UserForm



**Figur 22: VBA Prosjekt Explorer**

**Kildekode**

HPFaktura.frm

-----  
Public datoOprtLogg As String  
Public datoAvslLogg As String  
Public vlgkKnr As Integer  
Public vlgkLoggId1 As String  
Public endretStatus As String

Private Sub calAvslLogg\_Click()

Dim dag As String  
Dim mnd As String  
Dim aar As String

dag = calAvslLogg.Day  
mnd = calAvslLogg.Month  
aar = calAvslLogg.Year

datoAvslLogg = dag & "." & mnd & "." & aar

End Sub

Private Sub calOprttLogg\_Click()

Dim dag As String  
Dim mnd As String  
Dim aar As String

dag = calOprttLogg.Day       'Henter ut dag fra calender  
mnd = calOprttLogg.Month     'Henter ut måned fra calender  
aar = calOprttLogg.Year      'Henter ut år fra calender

datoOprtLogg = dag & "." & mnd & "." & aar

End Sub

Private Sub cmbVlgKnr\_Click()

vlgkKnr = cmbVlgKnr.Value  
cmbVlgKnr.Text = " " & vlgkKnr       'Skriver valgt kundenummer  
til cmbVlgKnr

End Sub

Private Sub cmbVlgKnr\_DropButtonClick()

cmbVlgKnr.Clear               'Rensker listen for forrige verdier  
Dim sqlString As String       'Variabel for Sql strengen  
Dim rst As recordSet         'ADO Recordset object'et  
Dim rstField

## Vedlegg - kildekode

```
'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

sqlString = "SELECT Kundenummer FROM Kunde"

Set rst = dbCon.Execute(sqlString)

'Lukker databasen
Set dbCon = Nothing
'dbCon.Close

If Not rst.EOF Then                                'Sjekker om slutten av rst

Do While Not (rst.EOF)

For Each rstField In rst.Fields

cmbVlgKnr.AddItem (rstField) 'Skriver Kundenummer til cmbVlgKnr
Next

rst.MoveNext                                     'Flytter til neste field i rst

Loop

End If

End Sub

Private Sub cmbEndreStatus_Click()

endretStatus = cmbEndreStatus.Value

cmbEndreStatus.Text = " " & endretStatus          'Skriver ny valgt
status til cmbEndreStatus

End Sub
```

## Vedlegg - kildekode

```
Private Sub cmbEndreStatus_DropButtonClick()

    cmbEndreStatus.Clear                'Rensker listen for forrige
    verdier
    Dim sqlString As String              'Variabel for Sql strengen
    Dim rst As recordSet                 'ADO Recordset object'et
    Dim rstField

    'Åpner databasen
    Dim dbCon As ADODB.Connection
    Set dbCon = New ADODB.Connection

    'Set the Connectionstring
    'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
    WLAN;uid=sa;pwd=;database=HP_faktura"

    'Alternative Connectionstring
    dbCon.ConnectionString = "DSN=HP_faktura"

    'Open a Connection to the database
    dbCon.CommandTimeout = 15
    dbCon.Open

    sqlString = "SELECT Status FROM Status"

    Set rst = dbCon.Execute(sqlString)

    'Lukker databasen
    Set dbCon = Nothing
    'dbCon.Close

    If Not rst.EOF Then                  'Sjekker om slutten av rst

        Do While Not (rst.EOF)

            For Each rstField In rst.Fields

                cmbEndreStatus.AddItem (rstField) 'Skriver Kundennummer til
                cmbVlgKnr
            Next

            rst.MoveNext                  'Flytter til neste field i rst

        Loop

    End If

End Sub

Private Sub cmbVlgKonsulent_Click()
```

## Vedlegg - kildekode

```
'Dim totNavn As String
totNavn = cmbVlgKonsulent.Value
cmbVlgKonsulent.Text = " " & totNavn

End Sub

Private Sub cmbVlgKonsulent_DropButtonClick()

cmbVlgKonsulent.Clear                'Rensker listen for forrige
verdiier
Dim sqlString As String              'Variabel for Sql strengen
Dim rst As recordSet                'ADO Recordset object'et
Dim rstField
Dim fNavn As String
Dim eNavn As String
Dim navn As String

'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

sqlString = "SELECT Fornavn, Etternavn FROM Konsulent"

Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then                'Sjekker om slutten av rst

Do While Not rst.EOF
fNavn = rst.Fields("Fornavn")
eNavn = rst.Fields("Etternavn")
navn = fNavn & " " & eNavn        'Setter sammen navnestreng
cmbVlgKonsulent.AddItem (navn)    'Skriver konsulentnavn til
cmbVlgKonsulent
rst.MoveNext                        'Flytter til neste field i konsulentRst
Loop

End If

'Lukker databasen
```



## Vedlegg - kildekode

```
Set dbCon = Nothing
'dbCon.Close

End Sub

Private Sub cmbVlgLoggId1_Click()

Dim vlgLiggId As Integer
vlgLoggId1 = cmbVlgLoggId1.Value
cmbVlgLoggId1.Text = " " & vlgLoggId1

End Sub

Private Sub cmbVlgLoggId1_DropButton_Click()

cmbVlgLoggId1.Clear           'Rensker listen for forrige
                              verdier
Dim sqlString As String      'Variabel for Sql strengen
Dim rst As recordSet        'ADO Recordset object'et
Dim rstField

'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

sqlString = "SELECT Loggid FROM Prosjekt"

Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then          'Sjekker om slutten av rst

Do While Not (rst.EOF)

For Each rstField In rst.Fields

cmbVlgLoggId1.AddItem (rstField) 'Skriver Kundenummer til
cmbVlgKnr

Next
```

## Vedlegg - kildekode

```
rst.MoveNext                'Flytter til neste field i rst

Loop

End If

'Skriver vlgLoggId1 sin status til Prosjektstatus feltet
Dim vlgLoggId As Integer
vlgLoggId = Val(cmbVlgLoggId1.Value)

'Henter inn gjeldende loggid sin status fra db
sqlString = "SELECT Prosjektstatus FROM Prosjekt WHERE Loggid = ""
& vlgLoggId & """"
Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then

txtbPrsjStatus2.Text = rst.Fields.Item(0)

End If

'Lukker databasen
Set dbCon = Nothing
'dbCon.Close

End Sub

Private Sub cmdAvslLogg_Click()

Dim sqlString As String          'Variabel for Sql strengen
Dim maxLoggnummer As Integer
Dim gjeldendeLoggid As Integer
Dim rst As recordSet            'ADO Recordset object'et
Dim rstField
Dim Ansattnummer As Integer

'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

'Avslutter loggen ved å finne høyeste Loggnummer,
```

## Vedlegg - kildekode

'Skriver nemlig alltid til høyeste Lognummer.

'Må først plukke ut valgte konsulent sitt ansattnummer fo å kunne  
avlsutte loggen  
'på riktig konsulent

Dim navn As String

konsulentnavn = cmbVlgKonsulent.Value           'Plukker ut navnet p  
valgte konsulent

sqlString = "SELECT Fornavn, Etternavn FROM Konsulent"  
Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then                               'Sjekker om slutten av rst

Do While Not rst.EOF

fNavn = rst.Fields("Fornavn")  
eNavn = rst.Fields("Etternavn")  
navn = " " & fNavn & " " & eNavn

rst.MoveNext

If (navn = cmbVlgKonsulent.Value) Then 'Sjekker om valgte  
konsulent stemmer med Fornavn og Etternavn i DB

sqlString = "SELECT Ansattnummer FROM Konsulent WHERE Fornavn  
LIKE '" & fNavn & "' AND Etternavn LIKE '" & eNavn & "' "  
Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then                               'Henter i så fall ut Ansattnummer

Do While Not rst.EOF

Ansattnummer = rst.Fields.Item(0)  
rst.MoveNext

Loop

End If

End If

Loop

End If

'Henter ut høyeste lognummer på gjeldende ansattnummer  
gjeldendeLoggid = cmbVlgLoggId1.Value

## Vedlegg - kildekode

```
sqlString = "SELECT MAX(Loggnummer) FROM Prosjekt WHERE Loggid  
= '" & gjeldendeLoggid & "' AND Ansattnummer = '" & Ansattnummer  
& "' "  
Set rst = dbCon.Execute(sqlString)
```

```
If Not rst.EOF Then
```

```
maxLoggnummer = rst.Fields.Item(0)
```

```
End If
```

```
'Så må Db oppdateres med dato for avsluttet logg for gjeldende  
Loggid, Loggnummer og Ansattnummer
```

```
sqlString = " UPDATE Prosjekt SET Closedato = '" & datoAvslLogg & "'  
WHERE Loggid = '" & gjeldendeLoggid & "' AND Loggnummer = '" &  
maxLoggnummer & "' AND Ansattnummer = '" & Ansattnummer & "' "  
dbCon.Execute (sqlString)
```

```
'Må nå opprette en ny logg. Det gjøres ved å duplisere forrige loggs  
'metadata inn i den nye loggen med ny Opendato
```

```
sqlString = " Select Kundenummer, Prosjektnavn, Opprettetdato,  
Prosjektstatus FROM Prosjekt WHERE Loggid = '" & gjeldendeLoggid &  
"' AND Loggnummer = '" & maxLoggnummer & "' AND Ansattnummer  
= '" & Ansattnummer & "' "  
Set rst = dbCon.Execute(sqlString)
```

```
'Henter nå ut metadata fra loggen (recordsettet) og lagrer de i  
"dummyvariabler"
```

```
Dim dummyLoggid As Integer
```

```
Dim dummyKundenr As Integer
```

```
Dim dummyPrsjnavn As String
```

```
Dim dummyDato As String
```

```
Dim dummyPrsjStatus As String
```

```
If Not rst.EOF Then
```

```
'Do While Not rst.EOF
```

```
dummyKundenr = rst.Fields.Item(0)
```

```
dummyPrsjnavn = rst.Fields.Item(1)
```

```
dummyDato = rst.Fields.Item(2)
```

```
dummyPrsjStatus = rst.Fields.Item(3)
```

```
'Loop
```

```
End If
```

```
'Oppretter nytt loggnummer
```

```
maxLoggnummer = maxLoggnummer + 1
```

## Vedlegg - kildekode

```
'Oppretter nå ny logg
sqlString = " INSERT INTO Prosjekt (Kundenummer, Prosjektnavn,
OpprettetDato, Prosjektstatus, Loggnummer, Ansattnummer,
Opendato) VALUES ('" & dummyKundenr & "', '" & dummyPrsjnavn &
"', '" & dummyDato & "', '" & dummyPrsjStatus & "', '" &
maxLoggnummer & "', '" & Ansattnummer & "', '" & datoAvslLogg & "')
"
dbCon.Execute (sqlString)

'Lukker databasen
Set dbCon = Nothing
'dbCon.Close

End Sub

Private Sub cmdEndreStatus_Click()

Dim sqlString As String
Dim rst As recordSet

'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

'Må konvertere vltLoggId1 til integer for å kunne skrive til DB
Dim konvLoggId1 As Integer
konvLoggId1 = Val(vltLoggId1)

sqlString = "UPDATE Prosjekt SET Prosjektstatus = '" & endretStatus &
"' WHERE Loggid = '" & konvLoggId1 & "'"

dbCon.Execute (sqlString)

'Oppdaterer Prosjektstatus feltet i GUI
sqlString = "SELECT Prosjektstatus FROM Prosjekt WHERE Loggid = '"
& konvLoggId1 & "' "
Set rst = dbCon.Execute(sqlString)

If Not rst.EOF Then
```

## Vedlegg - kildekode

```
txtbPrsjStatus2.Text = rst.Fields.Item(0)

End If

'Lukker databasen
Set dbCon = Nothing
'dbCon.Close

End Sub

Private Sub cmdgenererId_Click()

Dim prsjBeskr As String
Dim prsjNavn As String
Dim sqlString As String
Dim prsjIdStatusrst As recordSet

prsjBeskr = txtbPrsjBeskr1.Text
prsjNavn = txtbPrsjNavn.Text
Dim gLogg As genLogg
Set gLogg = New genLogg
Call gLogg.genererLogg(datoOprtLogg, vlgtKnr, prsjBeskr, prsjNavn)

'Åpner databasen
Dim dbCon As ADODB.Connection
Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

sqlString = " SELECT MAX(Loggid) FROM Prosjekt "

Set prsjIdStatusrst = dbCon.Execute(sqlString)

'Skriver ut LoggId og Status på prosjektet
If Not prsjIdStatusrst.EOF Then

Dim nyLoggId As Integer
nyLoggId = prsjIdStatusrst.Fields.Item(0)
txtbLoggId.Text = nyLoggId
```

## Vedlegg - kildekode

```
sqlString = "SELECT Prosjektstatus FROM Prosjekt WHERE Loggid = ""  
& nyLoggId & """"  
Set prsjIdStatusrst = dbCon.Execute(sqlString)  
txtbPrsjStatus1.Text = prsjIdStatusrst.Fields.Item(0)  
  
End If  
  
'Lukker databasen  
Set dbCon = Nothing  
'dbCon.Close  
  
End Sub  
  
Private Sub cmdReg_Click()  
  
Set rA = New registrerAktivitet  
rA.Show  
  
End Sub  
  
Private Sub cmdRegistrer_Click()  
Dim Knr As Integer  
Dim Knvn As String  
Dim KTlfnr As String  
'  
  
Dim Fadr As String  
Dim FPostnr As String  
Dim FSted As String  
'  
  
Dim Badr As String  
Dim BPostnr As String  
Dim BSted As String  
'  
  
Dim RefPers As String  
Dim RefEpost As String  
Dim RefTlfnr As String  
  
'Henter ut verdier fra Kundeopplysninger sine respektive rammer  
'Kundeinformasjon  
Knr = Val(txtbKnr.Text)  
Knavn = txtbKnavn.Text  
KTlfnr = (txtbKTlfnr.Text)  
  
'Fakturaadresse  
Fadr = txtbFadr.Text  
FPostnr = txtbFPostnr.Text  
FSted = txtbFSted.Text  
  
'Besoeksadresse  
Badr = txtbBadr.Text  
BPostnr = txtbBPostnr.Text  
BSted = txtbBSted.Text
```

## Vedlegg - kildekode

```
'Referansepersoner
RefPers = txtbRefPers.Text
RefEpost = txtbRefEpost.Text
RefTlfnr = txtbRefTlfnr.Text

Dim rKunde As regKunde
Set rKunde = New regKunde
Call rKunde.registrerKunde(Knr, Knvn, KTlfnr, Fadr, FPostnr, FSted,
Badr, BPostnr, BSted, RefPers, RefEpost, RefTlfnr)

'Resetter kundeinformasjonsfeltene til default verdi
txtbKnr = "Kundenummer"
txtbKnavn = "Kundenavn"
txtbKTlfnr = "Telefonnummer"

'Fakturaadresse
txtbFadr = "Gateadresse"
txtbFPostnr = "Postnummer"
txtbFSted = "Sted"

'Besøksadresse
txtbBadr = "Gateadresse"
txtbBPostnr = "Postnummer"
txtbBSted = "Sted"

'Referansepersoner
txtbRefPers = "Referanseperson"
txtbRefEpost = "E-Post"
txtbRefTlfnr = "Telefon"

End Sub
-----

registrerAktivitet.frm
-----

Private Sub lstTaskId_Click()

If lstTaskId.Text = "RE (Revenue)" Then

lblAntTimer.Visible = True
txtbAntTimer.Visible = True
lblReiseSone.Visible = False
txtbReiseSone.Visible = False
lblTimeSats.Visible = True
txtbTimeSats.Visible = True
lblKost.Visible = False
txtbKost.Visible = False
lblSum.Visible = True
txtbSum.Visible = True
lblTotalSum.Visible = True
txtbTotalSum.Visible = True
```



## Vedlegg - kildekode

End If

If lstTaskId.Text = "TR (Training)" Then

```
lblAntTimer.Visible = True
txbAntTimer.Visible = True
lblReiseSone.Visible = False
txbReiseSone.Visible = False
lblTimeSats.Visible = True
txbTimeSats.Visible = True
lblKost.Visible = False
txbKost.Visible = False
lblSum.Visible = True
txbSum.Visible = True
lblTotalSum.Visible = True
txbTotalSum.Visible = True
```

End If

If lstTaskId.Text = "CO (Cost)" Then

```
lblAntTimer.Visible = True
txbAntTimer.Visible = True
lblReiseSone.Visible = False
txbReiseSone.Visible = False
lblTimeSats.Visible = True
txbTimeSats.Visible = True
lblKost.Visible = False
txbKost.Visible = False
lblSum.Visible = True
txbSum.Visible = True
lblTotalSum.Visible = True
txbTotalSum.Visible = True
End If
```

If lstTaskId.Text = "TA (Travel)" Then

```
lblAntTimer.Visible = False
txbAntTimer.Visible = False
lblReiseSone.Visible = True
txbReiseSone.Visible = True
lblTimeSats.Visible = False
txbTimeSats.Visible = False
lblKost.Visible = True
txbKost.Visible = True
lblSum.Visible = True
txbSum.Visible = True
lblTotalSum.Visible = True
txbTotalSum.Visible = True
End If
```

End Sub

```
Private Sub UserForm_Activate()
```

```
lstTaskId.AddItem "RE (Revenue)"  
lstTaskId.AddItem "TR (Training)"  
lstTaskId.AddItem "CO (Cost)"  
lstTaskId.AddItem "TA (Travel)"
```

```
End Sub
```

```
-----  
genLogg.cls  
-----
```

```
Public Loggnummer As Integer
```

```
Public Sub genererLogg(xdato As String, xcmbVlgtKnr As Integer,  
txtPrsjBeskr1 As String, txtbPrsjNavn As String)
```

```
Dim sqlString As String  
Dim sqlGetLoggid As String  
Dim maxLoggidrst As recordSet  
Dim maxLoggid As Integer  
Dim prsjBeskr As String  
Dim prsjNavn As String  
Dim dato As String  
Dim vlgtKnr As Integer
```

```
'Endelig variabeltilordning for skrivning til databasen
```

```
dato = xdato  
vlgtKnr = xcmbVlgtKnr  
prsjBeskr = txtPrsjBeskr1  
prsjNavn = txtbPrsjNavn
```

```
'MsgBox dato & "og" & vlgtKnr & "og" & prsjBeskr & "og" & prsjNavn &  
""
```

```
'Åpner databasen  
Dim dbCon As ADODB.Connection
```

```
Set dbCon = New ADODB.Connection
```

```
'Set the Connectionstring  
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-  
WLAN;uid=sa;pwd=;database=HP_faktura"
```

```
'Alternative Connectionstring  
dbCon.ConnectionString = "DSN=HP_faktura"
```

```
'Open a Connection to the database  
dbCon.CommandTimeout = 15  
dbCon.Open
```

## Vedlegg - kildekode

```
sqlGetMaxLoggid = "SELECT MAX(Loggid) FROM Prosjekt"
'Brukes for å plukke ut største loggid

Set maxLoggidrst = dbCon.Execute(sqlGetMaxLoggid)
'Kjører sql setn. som plukker ut største logid

If Not maxLoggidrst.EOF Then

maxLoggid = maxLoggidrst.Fields.Item(0)           'Henter ut
nåværende gyldige Loggid

End If

maxLoggid = maxLoggid + 1                         'Genererer den
neste loggid'en
Loggnummer = 1                                   'Genererer første
loggnummer

'Setter sammen sql-strengen for å opprette prosjekt og logg
sqlString = "INSERT INTO Prosjekt (Loggid, Kundenummer,
Prosjektnavn, Opprettetdato, Prosjektstatus, Prosjektbeskrivelse,
Loggnummer, Opendato) VALUES ('" & maxLoggid & "', '" & vlgTKnr &
"', '" & prsjNavn & "', '" & dato & "', ' Opened ', '" & prsjBeskr & "', '" &
Loggnummer & "', '" & dato & "' )"

'-----
'dbCon.Execute sqlString 'Enkel måte å kjøre sql query på
'-----

Dim oCom As ADODB.Command
Set oCom = New ADODB.Command

'Definerer command obj. sine innstillinger
oCom.CommandType = adCmdText           'Forteller command obj at om
sql query
oCom.CommandText = sqlString           'sqlString tilordnes command obj
oCom.CommandTimeout = 30

'Definerer command obj. sine parametere
Dim oParam As ADODB.Parameter
Set oParam = New ADODB.Parameter

Set oParam = oCom.CreateParameter
oParam.Type = adVarChar                 'Setter parametertype til tekst
oParam.Size = 30                       'Maks lengden på parameterne som
overføres
oParam.Direction = adParamInput        'Definerer retningen på SQL
strengen
'oParam.Value = 0
oCom.Parameters.Append oParam          'Legger parameterne inn i
command obj
Set oParam = Nothing
```

## Vedlegg - kildekode

```
Set oCom.ActiveConnection = dbCon 'Oppretter connection mellom  
Command obj og DB
```

```
'Dim rst As ADODB.recordSet
```

```
oCom.Execute sqlString 'Eksekverer SQL strengen for  
prosjekt
```

```
'**Må vists skrive til Logg her **
```

```
'Lukker databasen  
Set dbCon = Nothing  
'dbCon.Close
```

```
End Sub
```

```
-----  
regKunde.cls
```

```
-----  
Public Sub registrerKunde(xtxtbKnr As Integer, xtxtbKnavn As String,  
xtxtbKTlfnr As String, xtxtbFadr As String, xtxtbFPostnr As String,  
xtxtbFSted As String, xtxtbBadr As String, xtxtbBPostnr As String,  
xtxtbBSted As String, xtxtbRefPers As String, xtxtbRefEpost As String,  
xtxtbRefTlfnr As String)
```

```
Dim msg As String
```

```
'Variabler for Kundeinformasjon  
Dim txtbKnr As String  
Dim txtbKnavn As String  
Dim txtbKTlfnr As String
```

```
'Variabler for Fakturaadresse  
Dim txtbFadr As String  
Dim txtbFPostnr As String  
Dim txtbFSted As String
```

```
'Variabler for Besoeksadresse  
Dim txtbBadr As String  
Dim txtbBPostnr As String  
Dim txtbBSted As String
```

```
'Variabler for Referanseperson  
Dim txtbRefPers As String  
Dim txtbRefEpost As String  
Dim txtbRefTlfnr As String
```

```
'Variabel for å holde sql-strengen  
Dim sqlString As String
```

```
'Åpner databasen
```

## Vedlegg - kildekode

```
Dim dbCon As ADODB.Connection

Set dbCon = New ADODB.Connection

'Set the Connectionstring
'dbCon = "driver={SQL Server};" & "server=IKT01-ET-
WLAN;uid=sa;pwd=;database=HP_faktura"

'Alternative Connectionstring
dbCon.ConnectionString = "DSN=HP_faktura"

'Open a Connection to the database
dbCon.CommandTimeout = 15
dbCon.Open

'Endelig variabeltilordning for skiving til databasen
txtbKnr = txtbKnr
txtbKnavn = txtbKnavn
txtbKTlfnr = txtbKTlfnr
txtbFadr = txtbFadr
txtbFPostnr = txtbFPostnr
txtbFSted = txtbFSted
txtbBadr = txtbBadr
txtbBPostnr = txtbBPostnr
txtbBSted = txtbBSted
txtbRefPers = txtbRefPers
txtbRefEpost = txtbRefEpost
txtbRefTlfnr = txtbRefTlfnr

'Tilordner sql-strengen til sql-variabelen
sqlString = "INSERT INTO Kunde (Kundenummer, Kundenavn,
Telefonnummer, Fakturaadresse, Fakturapostnummer,
Fakturapoststed, Besoeksadresse, Besoekspostnummer,
Besoekspoststed, Refperson, Refepost, Reftlfnr) VALUES (" & txtbKnr
& ", " & txtbKnavn & ", " & txtbKTlfnr & ", " & txtbFadr & ", " &
txtbFPostnr & ", " & txtbFSted & ", " & txtbBadr & ", " & txtbBPostnr
& ", " & txtbBSted & ", " & txtbRefPers & ", " & txtbRefEpost & ", "
& txtbRefTlfnr & ")"
```

## Vedlegg - kildekode

```
oCom.CommandTimeout = 30

'Definerer command obj. sine parametere
Dim oParam As ADODB.Parameter
Set oParam = New ADODB.Parameter

Set oParam = oCom.CreateParameter
oParam.Type = adVarChar      'Setter parametertype til tekst
oParam.Size = 30            'Maks lengden på parameterne som
overføres
oParam.Direction = adParamInput 'Definerer retningen på SQL
strengen
'oParam.Value = 0
oCom.Parameters.Append oParam 'Legger parameterne inn i
command obj
Set oParam = Nothing

Set oCom.ActiveConnection = dbCon 'Oppretter connection mellom
Command obj og DB

'Dim rst As ADODB.recordSet

oCom.Execute sqlString 'Eksekverer SQL strengen

'Lukker databasen
Set dbCon = Nothing
'dbCon.Close

End Sub
-----
```