

KOMPENDIUM



Høsten 2019

Informasjonssystemer

Dette er et kompendium for emnet TDT4175 Informasjonssystemer holdt ved NTNU høsten 2019. Kompendiet er basert på «Principles of Information systems» av Stair og Reynolds, «BPMN Method & Style» av Bruce Silver, kapittel 2.1 og 2.2 i «Model-based Development and Evolution of Information Systems: A Quality Approach» av John Krogstie og forelesningsnotatene (F).

Kapittel 1 – Introduksjon til informasjonssystemer

Introduksjon ^(F)

Den industrielle revolusjonen (1760-1840) markerte overgangen fra menneske- og dyrekraft til en økt bruk av maskiner i ulike bruksområder. Det ble en mekanisering vha vann og dampkraftverk. På tilsvarende måte har digitaliseringen markert overgangen fra analoge, mekaniske og papirbaserte løsninger, til elektroniske og digitale løsninger. **Digitalisering handler ikke om teknologi, men hvordan teknologiske utviklinger endrer måten organisasjoner utfører virksomhet.** Det handler ikke om transformasjon, men om tilpasning. Hvilken evne organisasjonen har til å tilpasse seg hurtige endringer som skyldes utvikling av nye digitale teknologier, vil avgjøre digitaliseringen av organisasjonen. Et eksempel på digitalisering er amazon, som var en av de første online bokforhandlerne som lot brukere bestille bøker fra enhver lokasjon i verden. Amazon har blitt en av de ledende teknologibedriftene fordi de har kontinuerlig fornyelse ved at de effektivt endrer deres virksomhetsmodell i møte med nye digitale teknologier.



Mennesker bruker informasjonssystemer hver dag. **Et informasjonssystem (IS) er et sett med sammenhengende komponenter som samler, behandler, lagrer og sprer data og informasjon, og det gir en tilbakemelding som hjelper organisasjoner til å oppnå deres mål (eks: øke profitt).** Introduksjonen av informasjonssystemer er likevel ikke alltid en suksess, for eksempel opplevde Nike at deres nye system ikke klarte å bestille riktige mengde produkter og leverte produkter sent til deres kunder. Dette resulterte i at fortjenesten ble 33 % lavere enn estimert. Noen av årsakene til slike tilfeller kan være at:

- Problemet som løses av et informasjonssystem er ikke alltid åpenbart
- Informasjonssystemer kan ikke introduseres og fjernes uten å påvirke omgivelsene
- Effektene av informasjonssystem er uforutsigbare

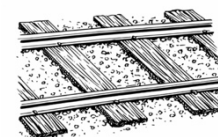
Viktig

Viktige konsepter ved informasjonssystem

Informasjon er en av de viktigste ressursene i en organisasjon. Det er viktig å skille mellom:

- **Data** – består av rå fakta, slik som ansattnummer, arbeidstimer, grafiske bilder, osv.
- **Informasjon** – samling av data som er ordnet og behandlet slik at det får mer verdi utenom verdien til de individuelle faktaene. For eksempel kan man se på summen av salgsdataen for å se totalt salg i måneden. Verdien til informasjonen som lages vil avhenge av relasjonene som defineres blant eksisterende data.

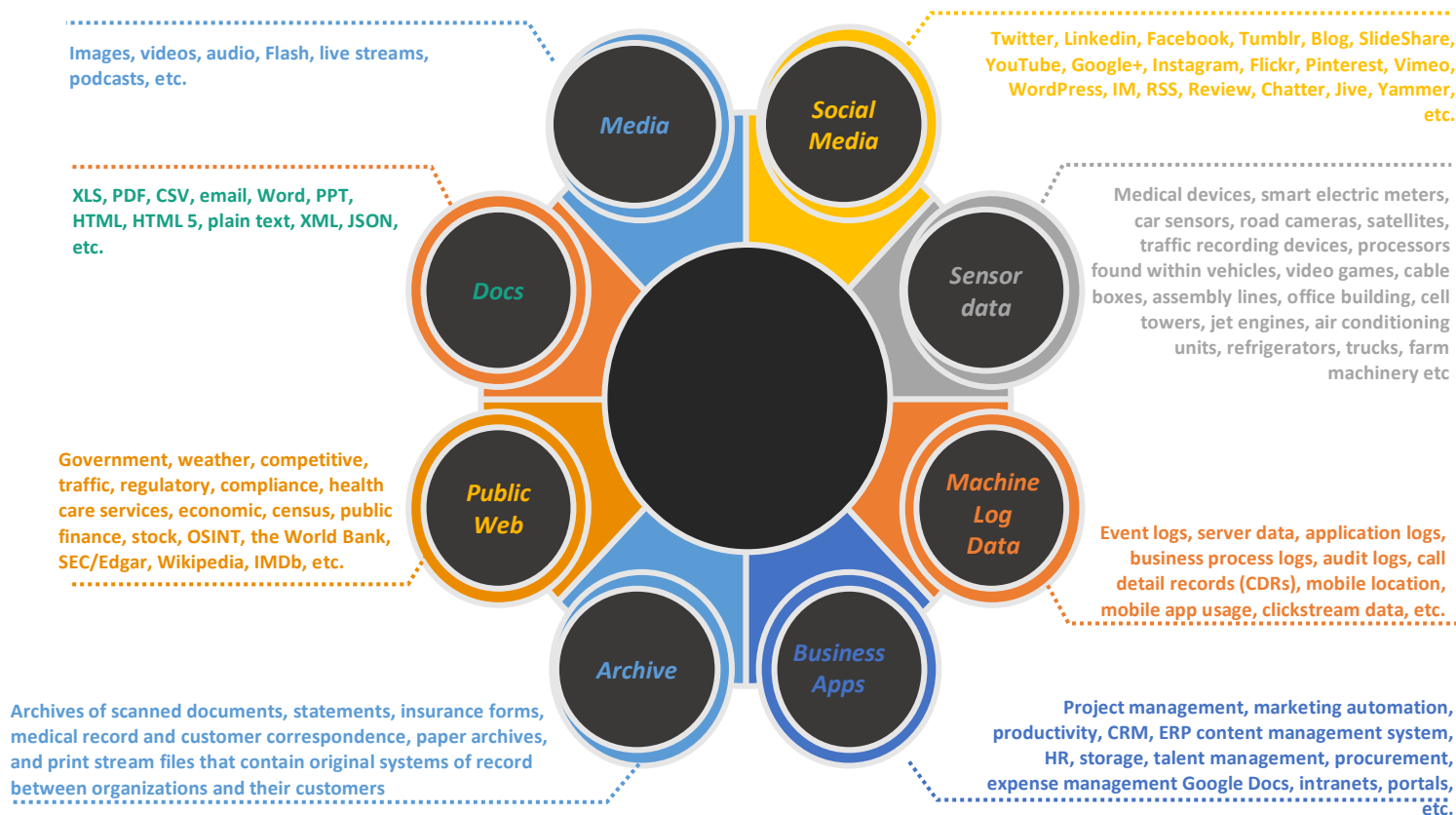
Man kan se på dataen som delene i et jernbanespor, der sporet representerer informasjonen. Hver del har begrenset inneboende verdi, men hvis man definerer et forhold mellom delene vil de utgjøre et fungerende jernbanespor som har større verdi. Ved å arrangere delene på en bestemt måte vil man få et bestemt jernbanespor. På samme måte vil type informasjon som lages avhenge av forholdet mellom dataen.



Andre konsepter er:

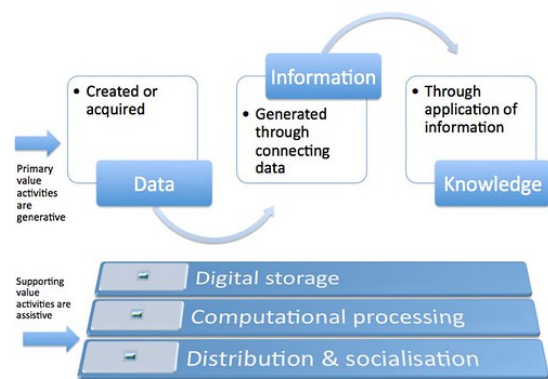
- **Prosess** – et sett med logisk relaterte oppgaver som utføres for å oppnå et definert utfall. Det er under prosessen at data blir omgjort til informasjon
- **Kunnskap** – bevissthet og forståelse av et sett med informasjon og hvordan denne informasjonen kan brukes for å støtte en bestemt oppgave. Det krever kunnskap for å definere forhold mellom dataen slik at man skaper nyttig informasjon

Informasjon er data som blir gjort mer nyttig ved å bruke kunnskap. I jernbanespor-analogien vil prosessen være selve byggingen av sporet, mens kunnskap er forståelsen av hvor mye plass sporet trenger, hvor mange tog som skal bruke det, osv.



Figuren over viser datakilder der ulike typer data stammer fra.

Figuren til høyre viser forholdet mellom data, informasjon og kunnskap. For eksempel kan dataen være 125, informasjonen er 125 kg eller at Per Hansen er 125 kg, mens kunnskap er at 125 kg betyr at Per Hansen har BMI over 30 og bør derfor slanke seg.



Viktig

Verdien til informasjon

Verdien til informasjon er direkte koblet til hvordan det hjelper beslutningstakere å oppnå organisasjonens mål. Verdifull informasjon kan gjøre det lettere for mennesker i organisasjonen å utføre oppgaver mer effektivt. Mange virksomheter antar at rapporter er basert på korrekt informasjon, men det er ikke alltid tilfellet. Ukorrekt data kan resultere i tap av potensielle nye kunder og redusert kundetilfredshet. Hvis informasjonen til organisasjonen ikke er korrekt eller fullstendig kan det føre til at ansatte gjør dårlige valg, noe som kan bli svært kostbart.

Egenskaper ved verdifull informasjon

Kvaliteten til et valg vil avhenge av kvaliteten til informasjonen som brukes for å ta dette valget. Kvaliteten til informasjonen kan defineres som hvorvidt dataen eller informasjonen er egnet bruken. Det er altså avhengig av konteksten og er subjektivt. Informasjonskvalitet er derfor multi-dimensjonalt, og den avhenger av følgende egenskaper:

- **Tilgjengelig** – informasjonen er lett tilgjengelig for autoriserte brukere, slik at de kan få tilgang til den ved behov
- **Nøyaktig** – informasjonen er feilfri. Hvis unøyaktig data brukes i transformasjonen vil det lages unøyaktig informasjon (GIGO: garbage in, garbage out)
- **Fullstendig** – informasjonen inneholder all viktig fakta.

- **Økonomisk** – informasjonen er relativt økonomisk å produsere, slik at beslutningstakerne kan balansere verdien og kostnaden ved informasjonen
- **Fleksibel** – informasjonen kan brukes til flere ulike formål
- **Relevant** – informasjonen er viktig for beslutningstakeren
- **Pålitelig** – informasjonen kan stoles på av brukerne, noe som ofte avhenger av en pålitelig samling av data.
- **Sikker** – informasjonen er sikret mot tilgang fra uautoriserte brukere
- **Enkel** – informasjonen er ikke unødvendig kompleks. For mye informasjon kan gjøre at beslutningstakeren ikke klarer å bestemme hva som er viktig
- **Tidsriktig** – informasjonen blir levert presist når det trengs
- **Verifiserbar** – informasjonen bør kunne sjekkes for å sikre at den er korrekt

Hvilke egenskaper som er viktigere enn andre vil avhenge av situasjonen og type valg som skal tas av beslutningstakerne. For eksempel for data med markedsinformasjon er det akseptabelt med noe unøyaktighet og ufullstendighet, men det er viktig at informasjonen er tidsriktig. For data som brukes i regnskap er det derimot essensielt at den er nøyaktig og fullstendig.

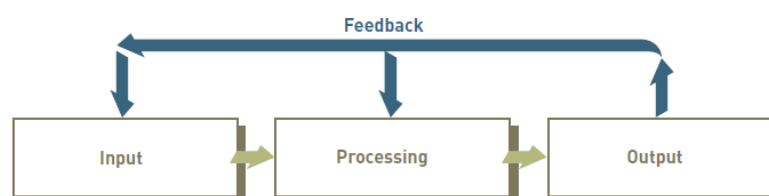
Hva er et informasjonssystem?

Viktig

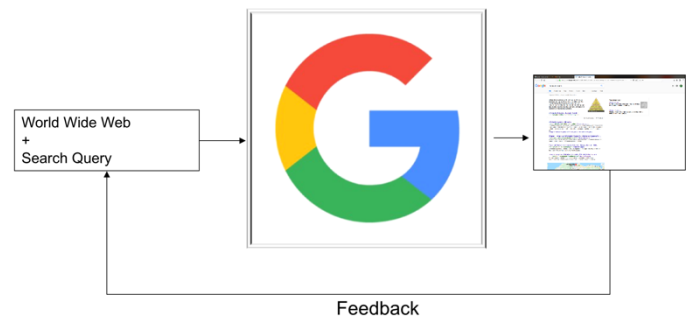
Vi interagerer med informasjonssystemer hver dag, både privat og i arbeid. For eksempel når vi får informasjon over internettet, samarbeider med kollegaer over mail, osv. **Et informasjonssystem er sammenhengende komponenter som arbeider sammen for å samle, prosessere, lagre og spre informasjon for å støtte beslutningstaking, koordinering, kontroll, analyse og visualisering i en organisasjon.** Informasjonssystemet gir en feedback-mekanisme for å kontrollere driften til systemet for å sikre at det fortsetter å oppnå målene. Denne mekanismen kan derfor hjelpe organisasjoner med å møte deres mål, for eksempel øke profitten eller forbedre kundeservicen.

Informasjonssystem består av grunnleggende komponenter som må settes sammen og integreres slik at de fungerer bra sammen. Figuren under viser de ulike komponentene ved et informasjonssystem:

1. **Input** – samlingen av rå data. For eksempel for å produsere lønninger må man først samle inn antall arbeidstimer per ansatt.
2. **Prosessering** – data blir transformert til nyttig output. Dette kan involvere beregninger, sammenligning av data, lagring av data for fremtidig bruk, osv. Det kan gjøres manuelt eller med dataassistanse. For eksempel må antall arbeidstimer hos hver ansatt gjøres om til netto lønn.
3. **Output** – produksjon av nyttig informasjon, som vanligvis er i form av dokumenter og rapporter. For eksempel kan det være lønnsjekken hos ansatte. Output fra et system kan bli input til et annet.
4. **Feedback** – informasjon fra systemet som brukes for å utføre endringer på input eller prosesseringsaktiviteter. For eksempel kan det ha skjedd en feil som gjorde at antall arbeidstimer til en ansatt ble 400 timer istedenfor 40. Informasjonssystem vil som regel sjekke at verdier er innenfor et logisk område, for eksempel 0-100 timer per uke. Systemet vil derfor bestemme at 400 er utenfor området og gi feedback om dette. Denne tilbakemeldingen brukes dermed for å sjekke og korrigere input. Feedback kan også brukes for å detektere flaskehals og øke produktiviteten.



Et eksempel på et informasjonssystem er søkemonitoren Google. Input er da spørsmålet og Internettet, prosesseringen er mekanismene i Google, mens output er de ulike lenkene som gis etter søket. I dette tilfellet kan feedback være hastighet (hvor raskt resultatet ble gitt), presisjon (hvor nyttig søkeresultatene er) og fullstendighet (hvor fullstendig er resultatet).

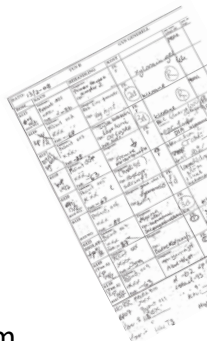


Computer-based informasjonssystem (CBIS)

Et informasjonssystem kan være manuelt (eks: papirbasert pasientliste) eller datastyrt (eks: Blackboard). **Et datastyrt informasjonssystem (CBIS = *Computer-Based Information System*) er et enkelt sett med maskinvare, programvare, databaser, nettverk, mennesker og prosedyrer som brukes for å samle, manipulere, lagre og prosessere data til informasjon.**



Stadig flere virksomheter bruker CBIS i deres produkter og tjenester, for eksempel produseres det biler med mulighet for avansert navigasjonssystem, ryggekameraer, osv. Figuren til venstre viser komponentene i en CBIS. **En CBIS er et enkelt sett med disse komponentene, mens alle settene utgjør den teknologiske infrastrukturen til virksomheten.** Den teknologiske infrastrukturen består altså av alle datastyrt informasjonssystem som brukes i virksomheten, og det danner grunnlaget der alle systemer og applikasjoner bygges.



De ulike komponentene i en CBIS er:

- **Maskinvare** – datautstyr som brukes for å utføre aktiviteter innenfor input, prosessering, lagring og output. For eksempel tastatur, mus, chips, minne, osv.
- **Programvare** – dataprogram som styrer operasjonene til en datamaskin. Dette inkluderer system programvare (eks: Microsoft Windows) som styrer grunnleggende operasjoner i datamaskinen og applikasjon programvare (eks: Microsoft Excel) som lar deg utføre spesifikke oppgaver.
- **Database** – organisert samling av fakta og informasjon.
- **Nettverk (telekommunikasjon)** – datamaskiner og utstyr kobles sammen i en bygning, i landet eller i verden for å tillate elektronisk kommunikasjon. Innenfor organisasjonen blir informasjon utvekslet over et intranett, mens et ekstranett lar mennesker få tilgang til autoriserte deler av intranettet.
- **Mennesker** – de som styrer, kjører, programmerer og vedlikeholder systemet. Gode systemer lar vanlige mennesker produsere ekstraordinære resultat.
- **Prosedyrer** – stegene som følges for å oppnå et spesifikt enderesultat. For eksempel kan det være stegene i en operasjon, vedlikehold eller sikkerhet. Gode prosedyrer gir hvordan man kan oppnå ønsket resultat ved å oppgi hvem som gjør hva og når og hva man gjør når noe går galt. Bruk at et CBIS innebærer å lage og følge mange prosedyrer.

Det viktigste elementet i et CBIS er menneskene, siden de utgjør forskjellen mellom suksess og fiasko i alle organisasjoner. Når mennesker har fått god opplæring og følger effektive prosedyrer, kan arbeidet gjøres raskere og billigere, ressurser blir bedre utnyttet og det er lettere å tilpasse seg endringer. Når prosedyrer er godt dokumenterte kan bedriften redusere opplæringskostnader og oppnå en kortere læringskurve.

Ulike typer informasjonssystem

De fleste organisasjoner bruker en rekke ulike informasjonssystem til forskjellige formål. Disse informasjonssystemene (IS) kan deles inn i tre ulike kategorier:

1. **Personlige IS** – informasjonssystem som forbedrer produktiviteten til individuelle brukere som utfører frittstående oppgaver. Et eksempel er programvare som brukes for å lage presentasjoner
2. **Gruppe IS** – informasjonssystem som forbedrer kommunikasjonen og støtter samarbeid blant medlemmer i en arbeidsgruppe. Et eksempel er programvare som brukes for å holde nettkonferanser.
3. **Enterprise IS** – informasjonssystem som brukes av organisasjoner for å definere strukturert interaksjon blant deres ansatte og/eller eksterne kunder, leverandører, offentlige etater, osv. Det kreves ofte radikalt redesign av grunnleggende arbeidsprosesser, for at slike system skal bli suksessfullt implementert. Et eksempel er system som behandler transaksjoner med leverandører.

	Personal IS	Group IS	Enterprise IS
Examples	Personal productivity software, decision-support system	Email, instant messaging, project management software	Transaction processing systems, enterprise systems, interorganizational systems
Benefits	Improved productivity	Increased collaboration	Increased standardization and ability to monitor work
Organizational complements (including well-trained workers, better teamwork, redesigned processes, and new decision rights)	<ul style="list-style-type: none"> Does not bring complements with it Partial benefits can be achieved without all complements being in place 	<ul style="list-style-type: none"> At least some complements must be in place when IS "goes live" Allows users to implement and modify complements over time 	<ul style="list-style-type: none"> Full complements must be in place when IS "goes live"
Manager's role	<ul style="list-style-type: none"> Ensure that employees understand and connect to the change Encourage use Challenge workers to find new uses 	<ul style="list-style-type: none"> Demonstrate how technology can be used Set norms for participation 	<ul style="list-style-type: none"> Identify and put into place the full set of organizational complements prior to adoption Intervene forcefully and continually to ensure adoption

Tabellen viser noen eksempler og egenskaper ved de ulike typene IS.

For hvert informasjonssystem som tas i bruk av en organisasjon, må følgende komponenter være på plass for at bruken skal lykkes:

- **Godt trente arbeidere** = ansatte må få god opplæring, og de må forstå behovet for det nye systemet, hva deres rolle er og hvordan de skal få ønsket resultat fra systemet
- **Støtte og infrastruktur for systemet** = opplærte og erfarne brukere kan vise andre hvordan man får verdi fra systemet
- **Godt teamarbeid** = ansatte må jobbe sammen for å oppnå de forventede fordelene
- **Redesignede prosesser** = nye systemer krever ofte radikal redesign av eksisterende arbeidsprosesser og automatisering av nye prosesser
- **Nye rettigheter og roller for beslutningstaking** = ansatte må forstå og akseptere deres nye roller og ansvar innenfor beslutningstaking.

Bedriftsinformasjonssystemer (Business IS)

Informasjonssystem kan brukes innenfor alle funksjonelle områder ved en bedrift, f. eks.:

- **Regnskap og finans** – forutsi renter og utgifter, kontrollere kontanter og andre finansressurser, analysere investeringer, osv.
- **Kundeservice** – samle data om kundene og deres interaksjon med bedriften for å bedre forstå deres behov og problemer.
- **HR** – skanne jobbsøknader, observere produktiviteten, osv.
- **Produksjon** – behandle bestillinger, kontrollere inventaret og følge med på produktkvaliteten.
- **Forskning og utvikling** – samle input fra kunder som fører til nye ideer
- **Salg** – bestemme reklamering og salgstilnærminger, sette optimale priser, osv.

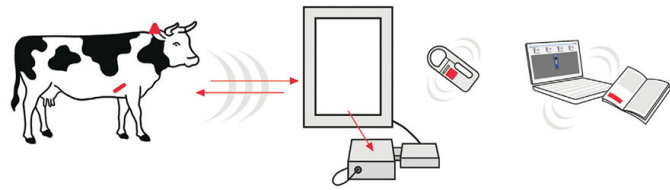
Profitability Planning	Financial Planning	Employment Planning, Outsourcing	Product Life Cycle Management	Sales Forecasting, Advertising Planning	STRATEGIC
Auditing, Budgeting	Investment Management	Benefits Administration, Performance Evaluation	Quality Control, Inventory Management	Customer Relations, Sales Force Automation	TACTICAL
Payroll, Accounts Payable, Accounts Receivable	Manage Cash, Manage Financial Transactions	Maintain Employee Records	Order Fulfillment, Order Processing	Set Pricing, Profile Customers	OPERATIONAL
ACCOUNTING	FINANCE	HUMAN RESOURCES	PRODUCTION/ OPERATIONS	MARKETING	

Tabellen viser bruksområdene for IS i en bedrift

Informasjonssystem blir også brukt innenfor nesten alle typer industri, som for eksempel:

- **Jordbruk** – bestemme hvor mye korn som skal høstes ved hvert jorde, forutsi utbytte slik at man kan bestemme hvilke jorder som bør gjødsles, osv.
- **Finans** – banker bruker informasjonssystem i avgjørelser innenfor lån og investeringer.
- **Helsevesen** – diagnostisere sykdommer, planlegge medisinske behandlinger, osv.
- **Gruvedrift** – globale posisjonssystem brukes for å identifisere og evaluere lovende områder for uthenting av mineraler
- **Husdyrhold** – RFID brukes for å samle data om bestemte kyr. For eksempel kan mengde og type fôr varieres avhengig av resultatet på målinger av melken de produserer

HOW LIVESTOCK RFID WORKS

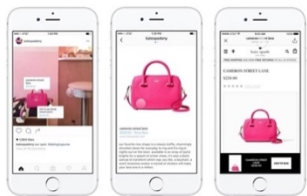
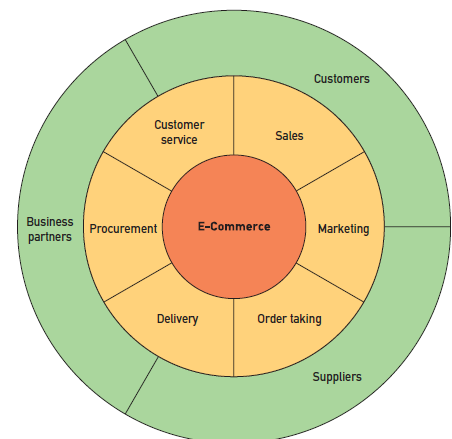


1. Apply ear tags to, and/or implant boluses in your animals
2. Read RFID identification codes to track individual animals using Datamars portable readers or raceway antennas
3. Use the included software to integrate data seamlessly to your livestock management system

Organisasjoner bruker flere ulike informasjonssystem for å forbedre hvordan de gjennomfører business og utfører faktabaserte beslutninger. Vi skal nå se på tre vanlige typer informasjonssystem som brukes av bedrifter: (1) E- og m-commerce, (2) enterprise systemer og (3) kunnskap management og spesialiserte IS.

1. Elektronisk og mobil handel (e- og m-commerce)

Elektronisk handel (e-commerce) involverer utveksling av penger for varer og tjenester over elektroniske nettverk. Alle bedrifter, uansett størrelse, får muligheten til å delta på det globale markedet siden elektronisk handel lar dem markedsføre og selge produkter eller tjenester over hele verden. E-commerce omfatter mange av prosessene som er utadrettet og involverer kunder, leverandører og andre business partnere (se figur til høyre). Dette kan for eksempel være salg, markedsføring, levering, kundeservice, osv.



Mobil handel (m-commerce) er bruken av mobile, trådløse enheter for å legge inn bestillinger og drive virksomhet. Eksempel på mobil e-commerce er appen til Sephora som er laget for å være lett og tilfredsstillende å bruke. Bruken av disse systemene skaper mange nye muligheter for bedrifter.

E-business går forbi e-commerce ved å bruke informasjonssystem og internett for å utføre alle bedriftsrelaterte oppgaver og funksjoner, slik som finans, markedsføring, HR, osv.

2. Enterprise system (TPS og ERP)

Enterprise system brukes for behandle daglige transaksjoner, der en transaksjon er enhver bedriftsrelatert utveksling, slik som betaling til ansatte og leverandører og levering av varer til kunder. Disse systemene lar bedriften utføre viktige oppgaver på enklere og mer effektive måter. I dag blir det fortsatt brukt tradisjonelle transaksjon prosesseringssystem (TPS) kombinert med management informasjonssystem (MIS), men disse blir i økende grad erstattet av ERP system.

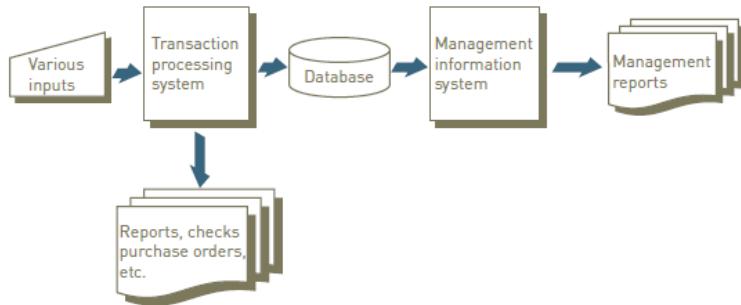
Transaksjons prosessering system (TPS) med management informasjonssystem (MIS)

Transaksjon prosesseringssystem (TPS) er et informasjonssystem som brukes for å utføre og registrere business transaksjoner. Med informasjonssystem menes en organisert samling av mennesker, prosedyrer, programvare, databaser og enheter. Lønnssystemet var et av de

første virksomhetssystemene som ble datastyrt (input: antall timer og timelønn, output: lønn). Andre eksempler er lagerkontroll, kundefakturering, osv.

Management informasjonssystem (MIS) er et informasjonssystem som brukes for å lage rapporter som gir rutineinformasjon til managere og beslutningstakere. Systemet bruker data og informasjon fra TPS for å lage standard rapporter, og det fokuserer på

effektiviteten ved driften. På figuren kan vi se hvordan TPS og MIS arbeider sammen for å prosessere bedriftstransaksjoner og lage management rapporter som kan brukes i videre beslutningstaking. TPS får input fra ulike kilder og behandler disse for å lage output og oppdatere en database med gyldige transaksjoner. MIS aksesserer denne databasen for å lage ulike rapporter.



Enterprise ressurs planlegging (ERP) – et enterprise system

Enterprise ressursplanlegging (ERP) er et sett med integrerte programmer som styrer den kritiske forretningsvirksomheten i en global organisasjon. Det er et enterprise system som i økende grad erstatter bruken av TPS og MIS. Systemet støtter rutineprosesser i virksomheten, opprettholder register om disse prosessene og gir omfattende rapportering og analyse av data. Det bruker én database

som kan deles av alle ansatte og evt. kunder og leverandører, slik at man eliminerer problemet der informasjon mangler eller er inkonsistent som følge av virksomheten bruker flere ulike TPS for hver avdeling. Et ERP system kan altså erstatte mange applikasjoner med ett sett med program, slik at systemet blir enklere å bruke og mer effektivt. Omfanget til et ERP system varierer, men de fleste gir integrert programvare som støtter produksjon og finans. ERP system kan også gi støtte for business analytics (mer senere) og e-business. SAP er et tysk selskap som leverer ERP programvare som er skreddersydd bedriftens drift. Tabellen viser hovedkomponentene ved en ERP for en produksjonsbedrift.

Module	Business Functions Addressed
Supply chain management	Manages all activities involved in sourcing and procuring raw materials, converting raw materials to finished product, warehousing, and delivering finished product to customers
Customer relationship management	Automates and integrates the sales, marketing, and customer service functions to capture and store customer and prospect contact information, account data, and sales opportunities in one central location
Product lifecycle management	Manages product information throughout the entire life cycle of a product from ideation, design and manufacture, through service and remaining product disposal—across all departments, contractors, and suppliers
Maintenance, repair, and operations	Automates and supports activities involved with the planning and scheduling of maintenance and repairs for any sort of mechanical, plumbing, or electrical device, along with the tracking of inventory and ordering of necessary parts and supplies
Accounting	Tracks the flow of data related to all the cash flows that affect an organization; manages functions related to setting up and maintaining the general ledger, accounts payable, accounts receivable, and payroll
Human resource management	Supports activities related to previous, current, and potential employees of the organization; provides tools for workforce analysis and planning, hiring, training, job and task assignment, performance evaluation, salary administration, managing employee benefits, retirement, and outplacement

Hvis ERP systemet implementeres riktig kan det gi følgende fordeler:

- Enklere å ta i bruk forbedret arbeidsprosesser
- Øker tilgang til tidsriktig data for beslutningstaking
- Eliminerer overflødige prosesser og assosiert infrastruktur
- Gir en global oversikt over operasjonell og planleggende data, slik at selskapet kan identifisere problemer og muligheter
- Automatisere kjerneoperasjoner
- Forbedre kundeservice



3. Kunnskap management og spesialiserte informasjonssystem

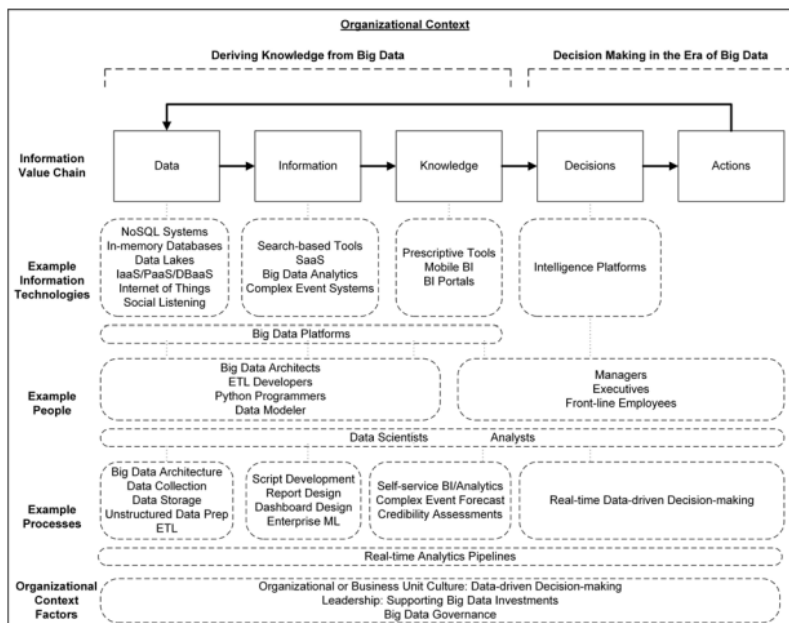
I tillegg til ERP, MIS og DSS kan bedrifter også bruke:

- **Kunnskap management system (KMS)** – et informasjonssystem som lagrer, henter og deler kunnskap, forbedrer samarbeid, lokaliserer kunnskapskilder, osv.. Med informasjonssystem menes en organisert samling av mennesker, prosedyrer, programvare, databaser og enheter. KMS blir ofte brukt i konsultentselskap for å dele kunnskap blant konsulentene, slik at hver konsulent blir mer verdifull og man slipper å løse samme problem fra starten av flere ganger. Et annet eksempel er i NASA, der KMS brukes for å fange den spesialiserte kunnskapen til de eldre ansatte før de slutter.
- **Spesialiserte informasjonssystem:**
 - **Kunstig intelligens (AI)** – det lages datasystemer med egenskaper som ligner menneskelig intelligens, slik at de kan ta over komplekse eller farlige rutineoppgaver.
 - **Ekspert systemer** – datamaskiner får evnen til å gi anbefalinger og fungere som en ekspert innenfor et bestemt felt, slik at det hjelper å øke ytelsen til den nye brukeren. De fanger og bruker visdommen til eksperter og spesialister, slik at den ikke går tapt når de slutter.

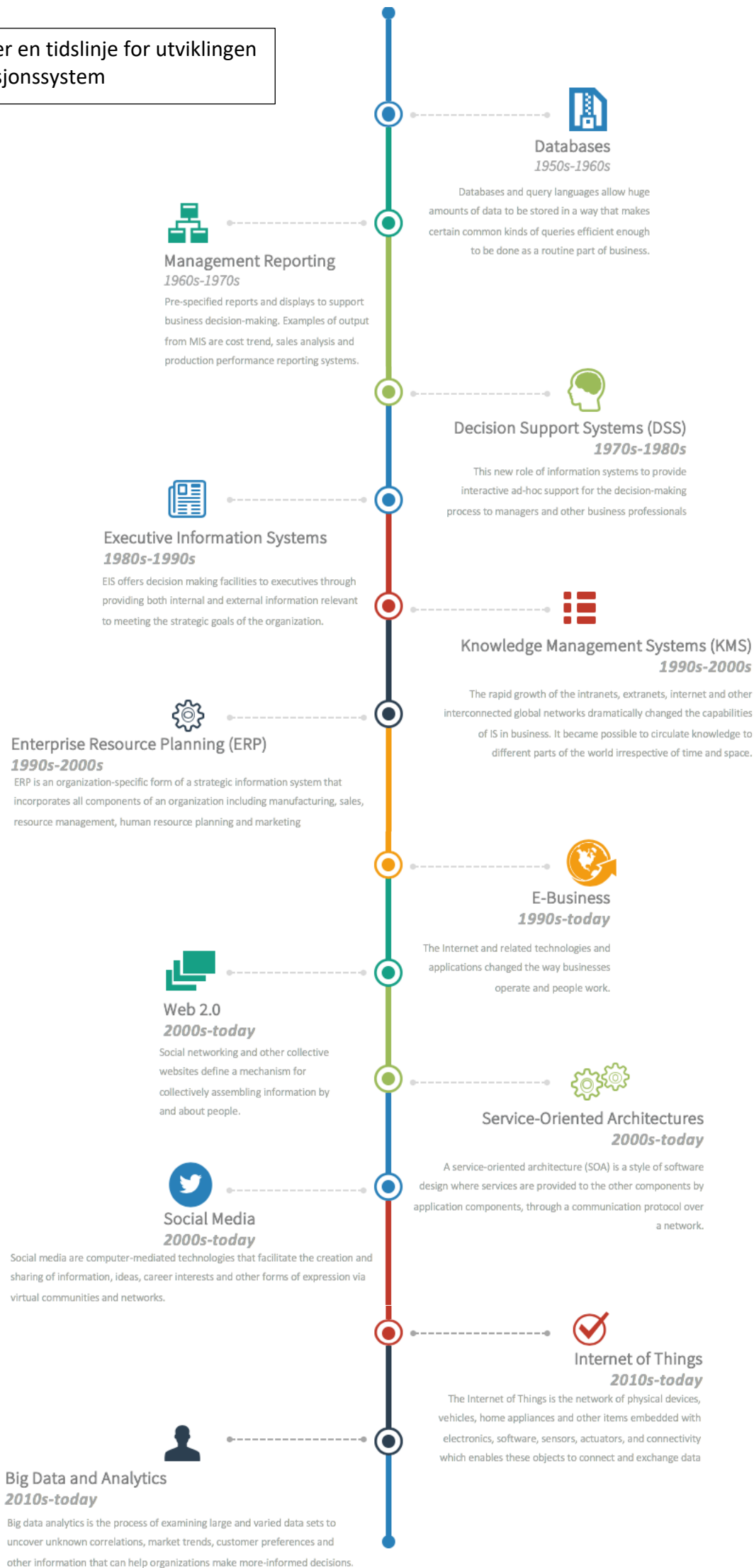
Business Intelligence og analytics

Ting, individer og organisasjoner genererer enorme mengde data som et biprodukt av deres aktivitet. Mange individer og organisasjoner bruker i tillegg tredjepart dataleverandører, slik som operatører i sosiale nettverk, for å få tilgang til mer data. For at denne dataen skal ha noe verdi, må den lagres, analyseres og rapporteres. Noen systemer som utnytter denne store mengden av data er:

- **Business Intelligence (BI)** = en rekke applikasjoner, praktiser og teknologier som brukes for uthenting, transformasjon, integrering, visualisering, analyse, tolkning og presentasjon av data for å støtte bedre beslutningstaking.
- **Business analytics** = omfattende bruk av data og kvantitativ analyse for å støtte faktabasert beslutningstaking i organisasjoner. Det kan brukes for å forstå ytelsen i bedriften, oppdage nye mønster og relasjoner, forklare resultat, osv. Det består av programvare som aksesserer, transformerer, lagrer, analyserer, modellerer og sporer informasjon, samt komponenter som kommuniserer resultatet til hele analysen.
- **Big data analytics** = prosessen av å undersøke store og varierte datasett for å finne ukjente korrelasjoner, markedstrender, kundepreferanser og annen informasjon som kan hjelpe organisasjonen til å ta mer velinformerte avgjørelser. Det er altså bruken av data med høy volum, mye variasjon og høy hastighet for å generere handlingsdyktig innsikt.



Figuren viser en tidslinje for utviklingen av informasjonssystemer



Strategisk planlegging og prosjektstyring

Strategisk planlegging og prosjektstyring er viktig for å sikre at organisasjonen arbeider effektivt på riktige prosjekter. Strategisk planlegging brukes for å forbedre samsvaret mellom behovene til bedriften og aktivitetene som gjennomføres av IS organisasjonen.

Denne planleggingen sikrer at IS organisasjonen og dens ressurser er fokusert på innsats som støtter hovedmålene til organisasjonen. En slik delt visjon vil fremme at IS organisasjonen ansetter de riktige menneskene med riktige egenskaper, velger riktige teknologier, installerer riktige system og fokuserer på prosjektene som trengs for at organisasjonen skal nå sine hovedmål. **God prosjektstyring er nødvendig for at en organisasjon skal få ønsket resultat for sin innsats.**

Systemanskaffelse og utvikling

Systemanskaffelse er prosessen som brukes for å få tak i informasjonssystemressurser som trengs for å gi tjenestene som er nødvendige for å tilfredsstille et sett med behov. Det er tre grunnleggende strategier for systemanskaffelse:

1. **Kjøpe eksisterende programvare** – ved å kjøpe eksisterende programvare som er utviklet av en programvare-produsent kan organisasjonen teste og evaluere programvaren før de gjør en stor forpliktelse ved å kjøpe og installere den. Programvaren kan installeres raskt, slik at brukerens behov blir raskt tilfredsstilt. Organisasjonen eier ikke programvaren og kan derfor ikke utføre endringer
2. **Bygge skreddersydd applikasjon** – kalles også systemutvikling og er en aktivitet der man bygger informasjonssystemet som møter brukerens behov. Prosjekter kan variere i størrelse, fra små til veldig store. Organisasjonen kan bruke egne ansatte til å utvikle systemet eller ansette et eksternt selskap.
3. **Velge en leverandør av programvaretjenester** – programvaren er en nettbasert tjeneste som brukere får tilgang til via skyen.

Organisasjonen betaler som regel en månedlig eller per-bruker avgift. Krever ingen installasjon, og leverandøren er ansvarlig for vedlikehold og oppgraderinger.

Strategy	Pros	Cons
Buy off-the-shelf software	+ A software solution can be acquired and deployed relatively quickly. + An organization can "test drive" software before acquiring it.	– Unmodified, the software may not be a good match to an organization's needs. – Maintenance and support costs can become excessive.
Build custom application	+ Customized software is more likely to be a good match to an organization's needs. + A custom application provides the potential to achieve competitive advantage.	– The cost to build a system can be quite high compared to the cost of purchasing off-the-shelf software. – Customizing software can mean it will be months or even years before the software solution is ready to deploy.
Choose a software service provider	+ Users do not need to purchase and install additional hardware or software. + The service provider handles necessary hardware and software maintenance and upgrades.	– Complex pricing arrangements and hidden costs may reduce expected cost savings. – Performance issues may cause wide variations in performance over time.

Informasjonssystemer i bedrifter og samfunnet

For at samfunnet, organisasjoner og individer skal kunne utnytte fordelene ved informasjonssystem, må disse systemene brukes forsiktig. Selv om informasjonssystemer kan gi enorme fordeler, innebærer de også mange problemer innenfor sikkerhet, personvern og etikk. Cyberkriminelle motiveres av potensialet for økonomisk gevinst. På verdensbasis anslås det at økonomiske tap som følge av nettkriminalitet er hele 400 milliarder dollar per år. Organisasjoner og land rundt om i verden blir også stadig mer bekymret for cyperterrorisme. For å lage et sterkt sikkerhetsprogram må organisasjonen vurdere hvilke trusler datamaskinene og nettverket er utsatt for og utdanne endebbrukere om risikoene som er involvert og handlingene de må utføre for å forhindre sikkerhetsbrudd.

Informasjonssystem har en stadig større rolle i samfunnet, næringslivet og industrien, og bruken av disse systemet fører til en rekke alvorlige etiske, juridiske og sosiale spørsmål. For eksempel har utviklingen av menneskelignende og sofistikerte system ført til at mennesker har mistet jobben og ulikheter i verden blir større som følge av ulik tilgang til datateknologi og internettet. Mange organisasjoner har utviklet etiske retningslinjer som hjelper ansatte når de skal ta vanskelige beslutninger.

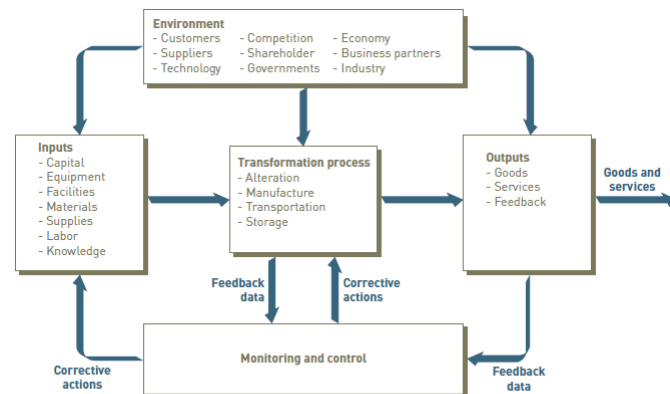
Kapittel 2 – Informasjonssystemer i organisasjoner

De seneste årene har informasjonssystemer endret hvordan organisasjoner arbeider. Initialt ble informasjonssystemer brukt for å automatisere manuelle prosesser, men etterhvert har de transformert arbeidets natur og formen til selve organisasjonene. Dette kapitlet ser på fordelene og problemene assosiert med bruken av IS i dagens organisasjoner.

Organisasjoner og informasjonssystemer

En organisasjon er en gruppe mennesker som er strukturert og styrt for å utføre et oppdrag eller tilfredsstille et sett med mål. Med strukturert menes det at det er definerte forhold mellom organisasjonens medlemmer og deres varierte aktiviteter. I tillegg er det definert prosesser som tildeler roller, ansvar og autoritet for å fullføre de ulike aktivitetene. Disse prosessene er ofte automatiserte vha informasjonssystem.

Organisasjoner er åpne system, noe som betyr at de påvirker og blir påvirket av omgivelsene (figur 1).

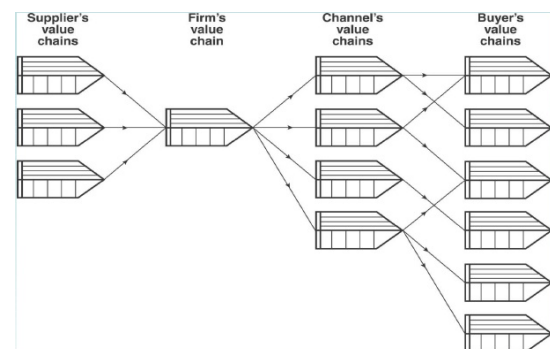


Figur 1: Generell modell av organisasjon

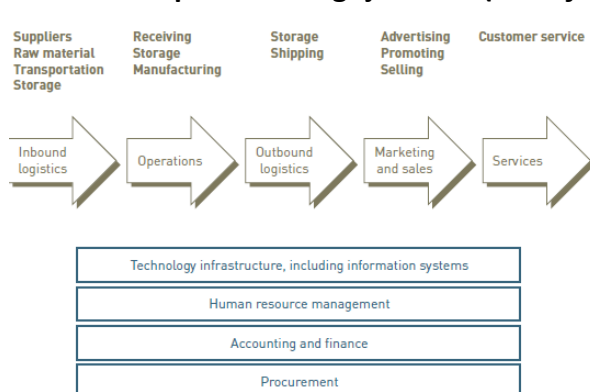
Viktig

Value og Supply Chains i organisasjoner

Hovedmålet til enhver organisasjon er å gi verdi til en interessent (kunde, leverandør, partner, osv.). Value chain brukes for å bestemme hvordan organisasjonen kan legge til verdi til deres produkter og tjenester, og begrepet ble først beskrevet av Michael Porter. **Value chain (verdikjede) er en rekke (kjede) aktiviteter som organisasjonen utfører for å transformere input til output på en slik måte at verdien til input økes.** En organisasjon kan ha flere verdikjeder og de sikrer deres suksess ved å legge til en signifikant mengde verdi til deres produkter og tjenester. Verdikjeden til en organisasjon vil ofte være koblet til verdikjeden til en annen organisasjon (se figur).



Supply chain (forsyningskjede) er en viktig verdikjede i organisasjoner som produserer produkter og tjenester (manufacturing organization). De viktigste aktivitetene i



forsyningskjeden er inngående logistikk, operasjoner, utgående logistikk, markedsføring og salg, og tjenester (se figur). For eksempel kan inngående logistikk involvere transport og lagring, mens operasjoner kan involvere selve produksjonen (se over pilene). Disse aktivitetene er direkte koblet til produksjon og levering av produkter eller tjenester, og de er avhengig av hverandre. Det er også fire hovedområder med støttende aktiviteter, inkludert teknologi infrastruktur, HR management, regnskap og finans og anskaffelse (*procurement*).

Supply Chain Management (SCM) innebærer alle aktivitetene som trengs for å få rett produkt i hendene til rette forbrukere i rett mengde ved rett tidspunkt og rett kostnad. SCM er hele prosessen fra anskaffelse av råmateriale til kundeleveranse.

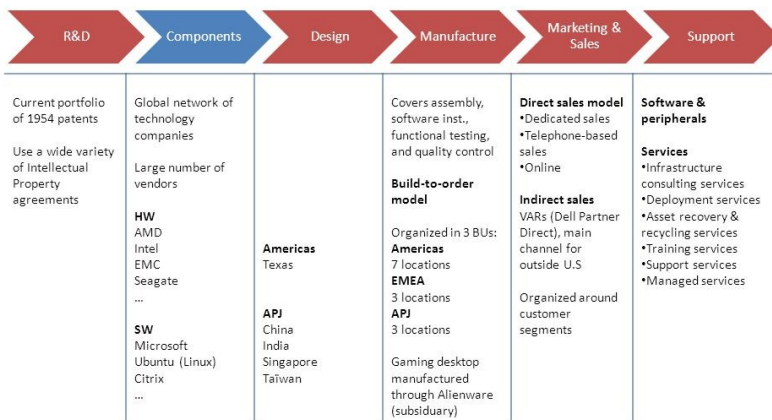
Eksempel – Dells Value Chain

Et eksempel på en verdikjede er den som brukes av Dell, som er et selskap som blant annet selger datamaskiner. Figurene viser ulike deler av deres verdikjede.

- Indirect distribution channel of the PC industry



- Dell's direct distribution channel



Red – heavy presence by Dell
Blue – limited presence by Dell

Supply Chains og informasjonssystem i organisasjoner

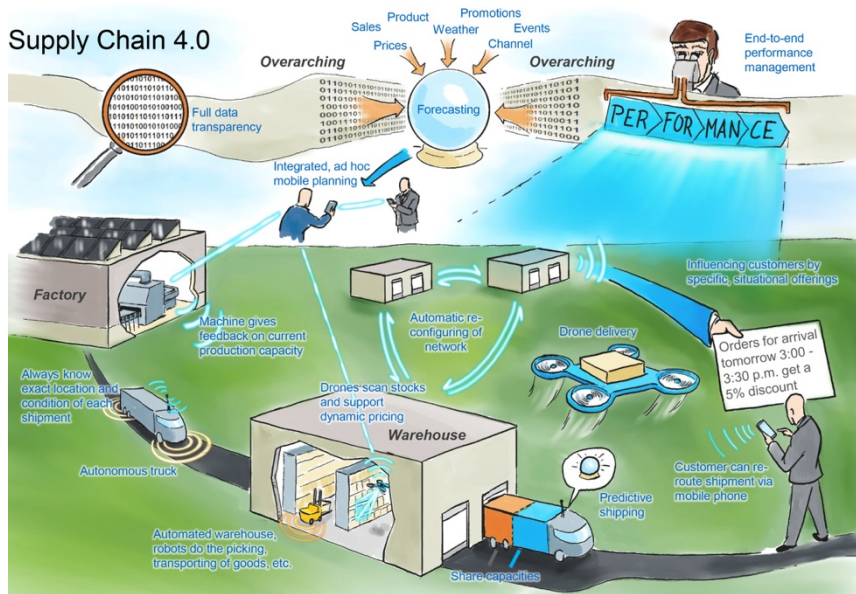
En supply chain kan involvere flere organisasjoner, og i slike tilfeller vil **organisasjonene være koblet sammen via fysisk flyt og informasjonsflyt**. Fysisk flyt involverer transformasjon, flytting og lagring av forsyninger og råmaterialer, mens informasjonsflyt lar deltager kommunisere deres planer og koordinere deres arbeid, samt styre den effektive flyten av varer og materialer opp og ned forsyningskjeden. Organisasjoner vil konstant tilpasse deres forsyningskjede, for eksempel er det mange selskap som tilbyr fri frakt til kunder i håp om at dette øker salget og profitten.

Supply chain integrering er en tett innretting og koordinering innenfor en supply chain. Det blir ofte brukt en delt management IS blant organisasjonene som er innblandet i supply chain. Det er flere nivåer ved denne integreringen, for eksempel kan første steg være å velge bestemte leverandører for å gi spesifikk input og utvikle en avtale med disse. Dette sikrer at selskapet har materialene som trengs for å utvikle output. Figuren viser fordeler med supply chain integrasjon, for eksempel kan det redusere mengde bortkastet materiale og tid, gjøre selskapet mer fleksibel for endring i markedet, redusere risikoen, gi større konkurransevne, osv.

Benefits of Supply Chain Integration



Supply Chain 4.0

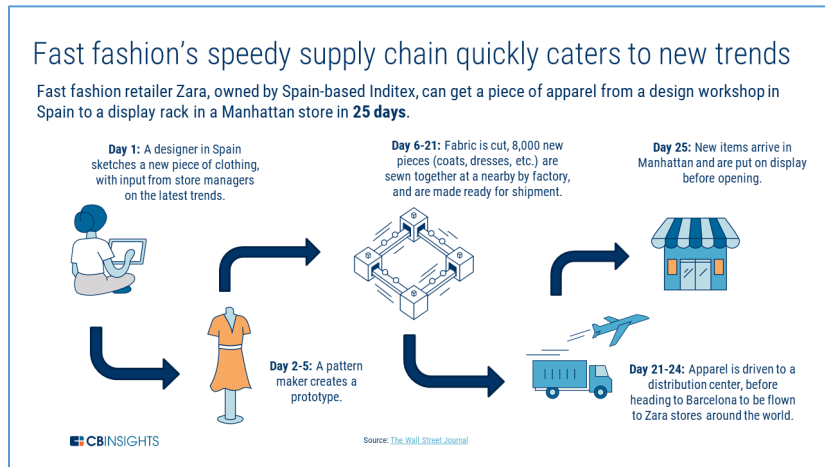


Informasjonssystem har forandret egenskapene til arbeidet og formen til selve organisasjonene. **De er ofte så involverte i aktivitetene som er en del av verdikjeden, at de blir en del av selve prosessen.**

Informasjonssystemer kan gi input, hjelpe til i transformasjonen av produkter eller produsere output (se figur). På neste side ser vi et eksempel på en organisasjon som bruker informasjonssystem som en del av supply chain.

Eksempel – Zaras supply chain

Zara er en klesbutikk som deltar i et miljø som er sterkt preget av konkurranse, både på pris og evne til å levere tilfredsstillende produkter til kundene. Derfor har Zara utviklet en svært responderende supply chain som lar dem gå fra design til salg i butikk på bare tre uker (gjennomsnitt er 6 måneder!). De nye produktene blir levert to ganger i uken til over 1600 butikker over hele verden. Mobile datamaskiner og salgspunkt-systemer brukes for å vurdere data fra butikker for å oppdage nye trender så tidlig som mulig. Denne dataen inkluderer informasjon om salg, inventar og kommunikasjon med kunde. Den blir analysert av designteam for å bestemme hva de skal designe videre og hvilke butikker som skal motta ulike varer. Zaras supply chain har ført til forbedret kundetilfredshet, redusert risiko for overflod av varer, reduserte kostnader og økt salg.



Endring i organisasjonen

Positive endringer er nøkkelen til en suksessfull organisasjon. Viktige begrep forbundet med endring er innovasjon, BPM, outsourcing, offshoring og downsizing.

Innovasjon

Innovasjon er bruken av nye ideer for produkter, prosesser og aktiviteter som fører til økt verdi, og det er en katalysator for vekst og suksess i enhver organisasjon. Uten innovasjon vil organisasjonen risikere å tape konkurransetrinnet og bli foreldet. Et eksempel er innovasjonen til Amazon der de forbedret sin supply chain ved å bruke roboter, investere i moderne varehus og effektivisere ordrebehandlingen for å tilby neste-dag og samme-dag levering av varer til kunden. Dette har gitt Amazon et midlertidig fortinn i konkurransen mot Google, eBay og Walmart. **Innovasjon kan føre til banebrytende produkter, nye inntektsstrømmer, økt fortjeneste og nye forretningsmodeller.**

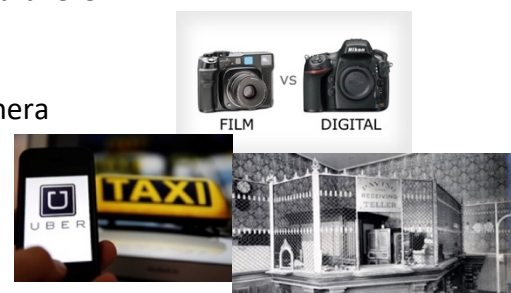
Innovasjon kan deles inn i to typer:

- **Opprettholdende innovasjon (*sustaining*)** – resulterer i forbedring av eksisterende produkter, tjenester og måter å operere. De lar organisasjonen kontinuerlig øke fortjenesten, redusere kostnader og få markedsandeler. Det kalles også inkrementell innovasjon. Et eksempel er utviklingen av iPhone (se figur).
- **Forstyrrende innovasjon (*disruptive*)** – en nyvinning som initialt gir en lavere ytelse enn det som er akseptert av markedet, men som over tid forbedres for å gi nye ytelsesegenskaper og bli mer attraktiv for brukere i et nytt marked. Ettersom innovasjonen blir bedre og gir en bedre ytelse vil den til slutt erstatte det tidligere produktet eller måten å gjøre ting på. Et eksempel er Netflix som erstatter kabel-TV for stadig flere brukere



Innovasjon kan også deles inn:

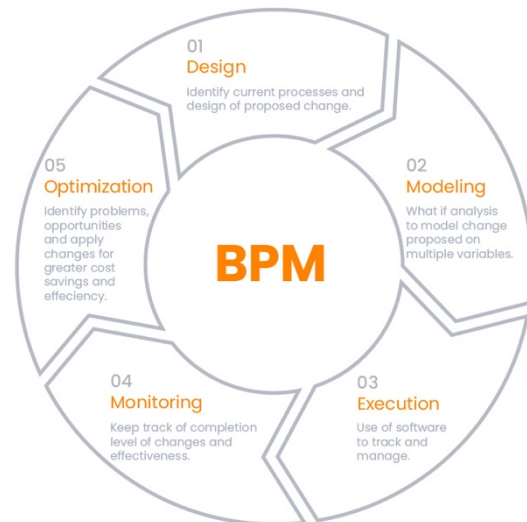
- **Produktinnovasjon** – for eksempel utvikling av digitale kamera
- **Tjenesteinnovasjon** – for eksempel utvikling av Uber
- **Prosessinnovasjon** – for eksempel utvikling av digitale tjenester innenfor bank og utbetaling av midler



Viktig

Business Process Management (BPM) (F)

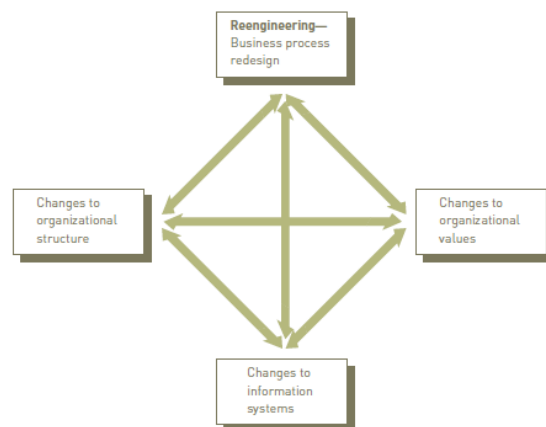
Business Process Management (BPM) går ut på å overvåke hvordan arbeid utføres i en organisasjon for å sikre konsistent utfall og utnytte forbedringsmuligheter. Det er en disiplin der mennesker bruker ulike metoder for å oppdage, modellere, analysere, måle, forbedre, optimalisere og automatisere business prosesser. Enhver kombinasjon av metoder som brukes for å kontrollere selskapets business prosesser er BPM.



Reengineering og kontinuerlig forbedring

For å være konkurransedyktige, må organisasjoner av og til utføre fundamentale endringer i måten de driver virksomheten. De må innovere og endre aktiviteter, oppgaver og prosesser som de bruker for å oppnå deres mål. To ulike måter å gjøre dette på er:

- 1. Business Process Reengineering (BPR)** – en form for innovasjon som involverer radikalt redesign av business prosesser, organisasjonsstrukturer, informasjonssystemer og verdier ved organisasjonen, og det brukes for å oppnå et gjennombrudd i resultatet til bedriften. Sterke handlinger brukes for å løse alvorlige problem. Det kan brukes for å redusere leveringstid, øke kvaliteten til produktet eller tjenesten, øke kundetilfredshet, osv.
- 2. Kontinuerlig forbedring** – en form for innovasjon som hele tiden leter etter måter å forbedre business prosesser og tilføre verdier til produkter og tjenester. Rutinehandlinger brukes for å utføre små forbedringer. Dette øker kundetilfredsheten og sikrer langsiktig fortjeneste.



Dette er altså to ulike aktiviteter innenfor Business Process Management (BPM). Tabellen viser en sammenligning av disse to fremgangsmåtene

Business Process Reengineering	Continuous Improvement
Strong action taken to solve serious problem	Routine action taken to make minor improvements
Top-down change driven by senior executives	Bottom-up change driven by workers
Broad in scope; cuts across departments	Narrow in scope; focuses on tasks in a given area
Goal is to achieve a major breakthrough	Goal is continuous, gradual improvements
Often led by resources from outside the company	Usually led by workers close to the business
Information systems are integral to the solution	Information systems provide data to guide the improvement team

Eksempel – BPR hos Ford Motors

Et eksempel på reengineering (BPR) er når Ford Motors utførte en radikal redesign i leverandøravdelingen for å kutte kostnader. Før redesignet bestod denne avdelingen av 500 ansatte, mens konkurrenten hadde bare 5. Målet til Ford var derfor å redusere antall ansatte og de brukte BPR for å effektivisere avdelingen.



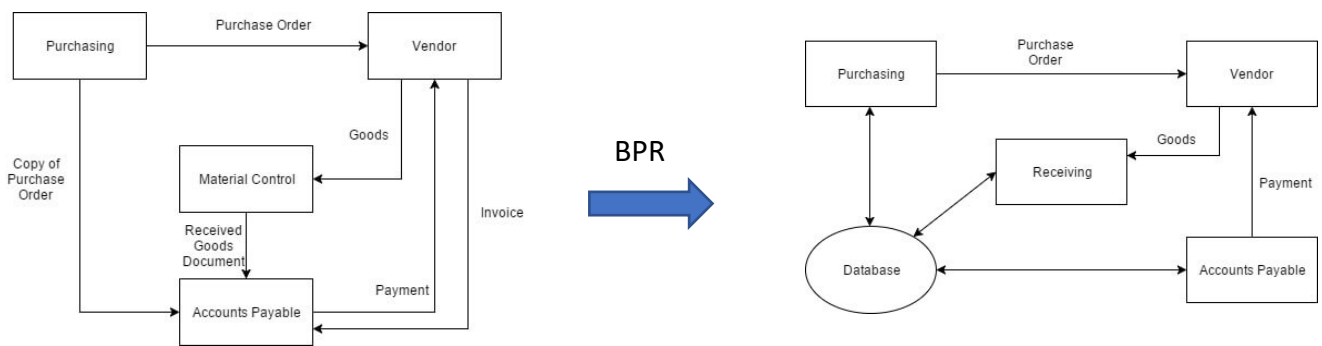
Prosessen før BPR

1. Innkjøpsavdelingen skriver en innkjøpsordre og sender en kopi til leverandøravdelingen
2. Materialkontrollen mottar varene og sender en kopi av dokumentet til leverandøravdelingen
3. Leverandøren sender en kvittering for varene til leverandøravdelingen
4. En ansatt ved leverandøravdelingen matcher de tre mottatte leveransene. Hvis de stemmer overens, vil betalingen utstedes.

Prosessen etter BPR

1. Innkjøp lager en bestilling som legges til en online database
2. Materialkontrollen mottar varene og utfører kryss-referanse med databasen for å sikre at det matcher en bestilling
3. Hvis det er en match, vil materialkontroll akseptere bestillingen på datamaskinen

Ingen behov for manuell matching av bestillinger!



Outsourcing, offshoring og downsizing (ikke i F)

En stor del av kostnadene i en organisasjon brukes for å ansatte, trene og kompensere ansatte, så det er viktig at organisasjonen kontrollerer antall ansatte som trengs for å opprettholde goder og tjenester med høy kvalitet. Strategier for å begrense disse er:

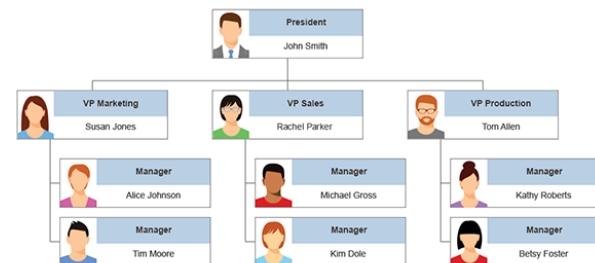
- **Outsourcing** = selskapet mottar tjenester fra en organisasjon utenfra som har ekspertise i å tilby den spesifikke funksjonen
- **Offshoring** = outsourcing der organisasjonen utenfra er i et annet land
- **Downsizing** = antall ansatte blir redusert for å kutte kostnader

Alle disse strategiene har assosierte etiske problemer og risikoer.

Organisasjonsstruktur

Organisasjonsstruktur er delenehetene i organisasjonen og hvilket forholdt de har til den totale organisasjonen.

Strukturen vil avhenge av målene til organisasjonen og dens tilnærming til management, og det kan påvirke hvordan organisasjonen bruker informasjonssystem. **De vanlige organisasjonsstrukturene er:**



- **Tradisjonell organisasjonsstruktur** – en hierarkisk og funksjonell struktur der organisasjonen er ordnet i en lederpyramide. På toppen er strategisk management som har størst autoritet i beslutningstakingen og størst innvirkning på målene til organisasjonen. Ved bunnen er ansatte som ikke har roller innenfor management. Et eksempel er militæret. Hvert nivå kontrollerer nivåene under og rapporterer til nivået over. Strukturen gir klare formelle relasjoner og hver ansatt har én manager, men det dannes skiller i organisasjonen og mange ansatte får ikke delta i beslutningstakingen.
- **Matrise organisasjonsstruktur** – et individ må rapportere til to managere der en er funksjonell (hvordan det gjøres) og den andre er operasjonell (hva skal gjøres). Dette blir ofte brukt i multinasjonale organisasjoner som opererer i flere områder av verden. De to managerne får fokusere på deres styrker, men det kan lettere oppstå konflikter, siden ansatte får beskjeder fra ulike hold.
- **Prosjekt organisasjonsstruktur** – strukturen er fokusert på hovedprodukter eller tjenester, med prosjektledere som styrer et eller flere prosjekter. Det er fokus på beslutningstaking og god kommunikasjon i team, men det blir mindre læring på tvers av team og mindre fokus på det totale målet til organisasjonen.
- **Virtuell organisasjonsstruktur** – en gruppe individer der medlemmene er geografisk spredt, men samarbeider ved å bruke informasjonssystem. Strukturen gjør at man kan bruke de beste tilgjengelige individene for å løse et viktig problem, men det kan bli vanskelig pga. forskjell i tidssone og mangel på ansikt-til-ansikt møter.

Organisasjonskultur og endringer

Informasjonssystem må implementeres slik at det passer godt inn i konteksten til en organisasjon og støtter dens fundamentale mål og strategier. **Organisasjonskultur er en samling av viktige forståelser og antagelser som deles av menneskene i organisasjonen.** Forståelsene kan være felles oppfatninger, verdier og tilnærminger til beslutningstaking, og de er som regel ikke dokumentert som mål eller formelle retningslinjer. For eksempel kan det innebære forventninger om at ansatte ofte sjekker mail i virtuelle teams.

Organisasjonsendring er hvordan organisasjonen planlegger, implementerer og håndterer endringer. Evnen til å effektivt introdusere endringer er essensielt for å kunne suksessfullt introdusere et informasjonssystem. Implementering av endring introduserer konflikter, forvirringer og forstyrrelser. Ansatte må stoppe å gjøre ting slik de er vant til og begynne å gjøre det på en ny måte. Suksessfull implementering av endring vil kun skje hvis ansatte aksepterer behovet for endringen og tror at endringen vil forbedre deres produktivitet. **Den myke siden av implementeringen innebærer arbeidet som trengs for å få ansatte til å omfavne det nye informasjonssystemet og den nye måten å arbeide på.** Dette er den største utfordringen for å suksessfullt implementere endringer, men det blir ofte oversett, noe som resulterer i at prosjektet feiler.

Endringsmodell (Change management model)

En endringsmodell kan brukes for å se hvordan endringen blir implementert. Den beskriver fasene som individet eller organisasjonen går gjennom når en endring blir gjennomført, og modellen gir prinsipper for å suksessfullt implementere endringen. To endringsmodeller som kan brukes for å håndtere den myke siden av implementeringen er Lewins endringsmodell og Leavitts diamant, som vi nå skal se nærmere på.

Lewins endringsmodell og Lewins Force Field Analysis

Lewins endringsmodell deler implementeringen av endringen i tre steg og identifiserer viktige oppgaver som må gjøres i løpet av hvert steg:

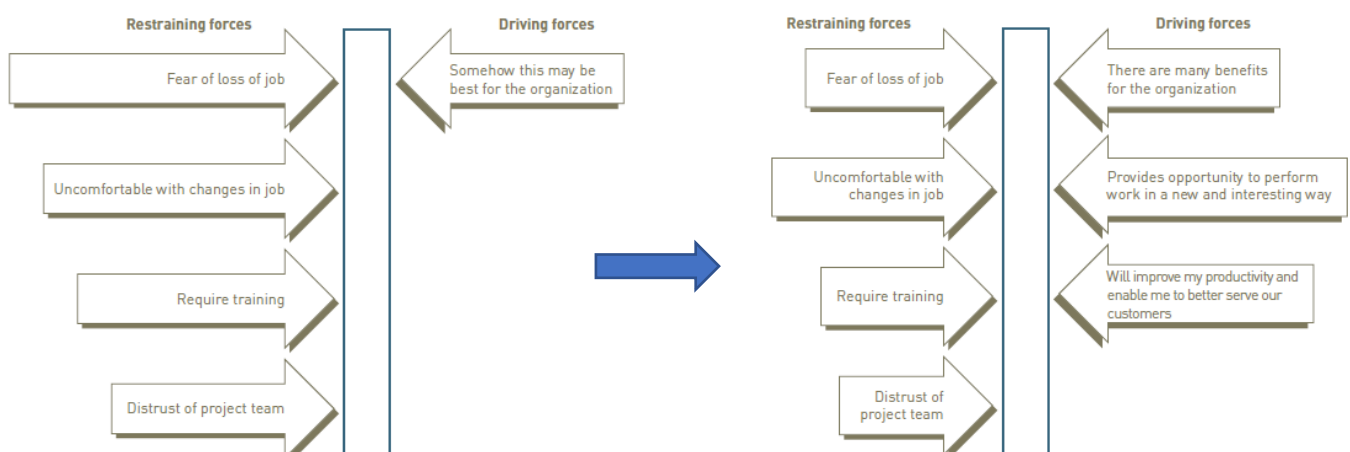
1. **Unfreezing** = forberede seg for endring og slutte med gamle vaner. Dette kan involvere kommunikasjon om hva, når, hvordan, tildeling av roller, osv.
2. **Moving** = utfører endringen ved å lære nye arbeidsmetoder og system. Dette kan involvere motivering, opplæring, management, feedback, osv.
3. **Refreezing** = institusjonalisering der endringene blir forsterket, slik at nye prosesser blir naturlig og en del av jobben til de ansatte. Dette kan involvere å følge med på progresjonen, belønne ansatte som viser ny oppførsel, osv.

Viktig

Lewins Force Field Analysis er en utvidelse av denne endringsmodellen, som inkluderer analyse av positive og negative krefter som avgjør om endringen kan skje:

- **Drivende kraft (positiv)** = oppfatninger, forventninger og kulturelle normer som har tendens til å fremme en endring
- **Begrensende kraft (negativ)** = gjør det vanskelig å akseptere en endring eller å arbeide for å implementere en endring

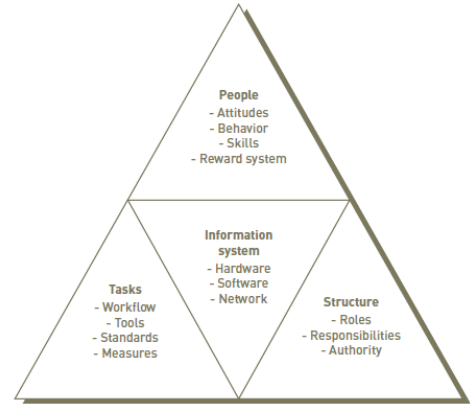
For at endring skal skje må styrken til drivende krefter overgå styrken til begrensende krefter. Dette kan gjøres ved å skape nye drivende krefter, øke styrken til eksisterende drivende krefter eller eliminere/svekke begrensende krefter. Figuren til venstre viser force-field analysen av en gruppe ansatte som nettopp har fått vite at et nytt IS har blitt installert. Lengden til pilen er styrken til kreftene. Figuren til høyre viser force-field analysen av samme gruppe etter at manageren har snakket med gruppen for å få de til å akseptere systemet.



Viktig

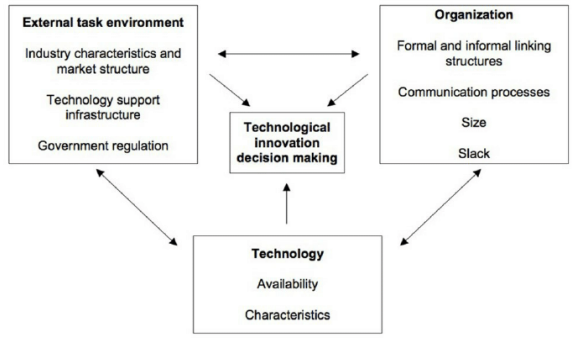
Leavitts diamant

Leavitts diamant er en endringsmodell som foreslår at alle organisasjonssystem er bygd opp av fire hovedkomponenter: mennesker, oppgaver, struktur og teknologi. Enhver endring i en av disse komponentene vil kreve passende endringer i de andre tre komponentene. For eksempel hvis man skal suksessfullt implementere et nytt informasjonssystem, må passende endringer også utføres blant menneskene, strukturen og oppgavene som påvirkes av det nye systemet. Menneskene må ha positiv innstilling, være villige til å vise ny oppførsel, lære nye egenskaper og få belønning for det. Strukturen må endres slik at man får nye roller med bestemt ansvar og autoritet. Oppgavene kan kreve nye verktøy og kvalitetsmålinger, og arbeidsflyten må oppdateres. **De største utfordringene ved implementering av et informasjonssystem er forbundet med atferd og ikke teknologi.** For å suksessfullt introdusere et nytt informasjonssystem kreves det derfor en blanding av egenskaper innenfor organisasjonsendring og det teknologiske. Sterkt og effektivt lederskap trengs for å overkomme atferdsproblemer og oppnå suksessfull introduksjon av systemet.



Organisasjonslæring

Organisasjonslæring handler om hvordan organisasjonen vil tilpasse seg nye betingelser eller endre praksiser over tid basert på erfaring og ideer.

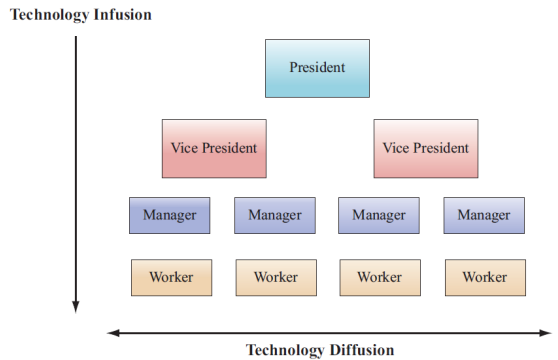


Ansatte lærer bedre måter å oppfylle deres rolle og inkorporerer dette i deres daglige aktiviteter. **Disse tilpasningene kan kreve reengineering (s. 14) eller resultere fra kontinuerlig forbedring. Technology Organization Environment (TOE) rammeverket beskriver faktorene som påvirker om organisasjonen vil tilpasse seg ny teknologi.** For eksempel vil størrelsen og kommunikasjonsprosesser i organisasjonen være avgjørende faktorer.

Brukertilfredshet – teknologisk diffusjon og infusjon (F)

Vi skiller mellom:

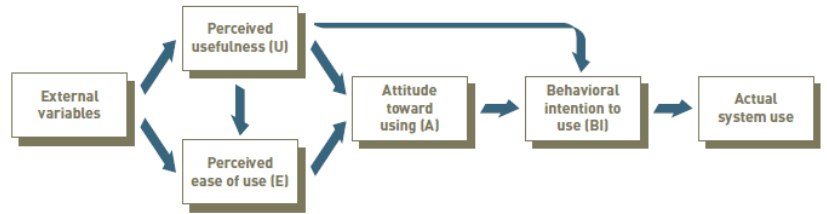
- **Teknologisk diffusjon** – hvor mye teknologien er spredt over en organisasjon
- **Teknologisk infusjon** – hvor mye teknologien gjennomsyrrer én avdeling med ansatte

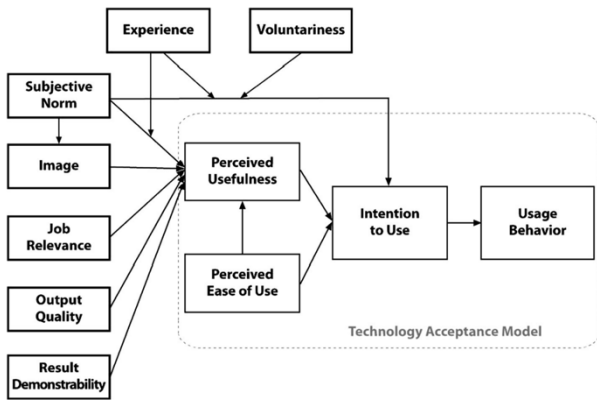


Viktig

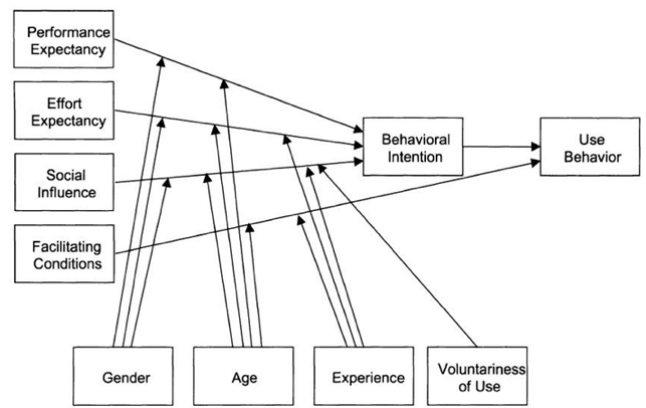
Brukertilfredshet og Teknologiakseptanse modell (TAM)

Teknologiakseptanse modell (TAM) gir faktorene som kan føre til bedre holdninger om bruken av et nytt informasjonssystem. Figuren viser en mulig TAM. *Perceived usefulness* er i hvilken grad individer tror at bruken av systemet vil forbedre deres ytelse. *Perceived ease of use* er i hvilken grad individer tror at systemet er lett å lære og bruke. Disse kan påvirkes av eksterne variabler, for eksempel meningene til andre individer. Sammen vil dette bestemme individers holdning til systemet og deres intensjon om å bruke systemet, som igjen påvirker den faktiske bruken. TAM er gode til å forklare hvorfor mennesker vil bruke et informasjonssystem eller en ny teknologi. De brukes i ulike kontekster, men kan bare forklare maksimalt 70% av akseptansen. Dette kan skyldes at noen faktorer er vanskelig å måle.



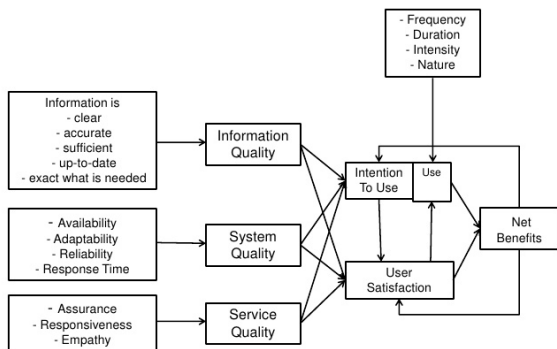


Figur 1; TAM som viser noen av de eksterne faktorer

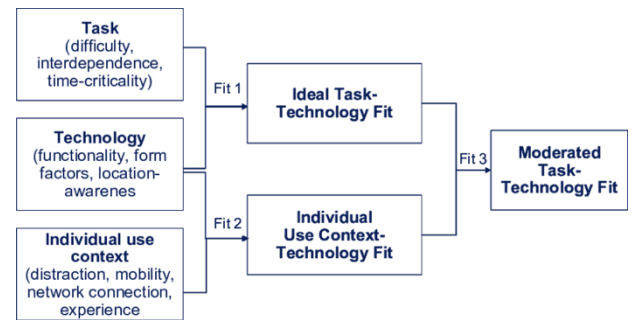


Figur 2: Unified Theory of Acceptance and Use of Technology (UTAUT). En TAM som brukes for å forklare individets intensjon om å bruke IS og etterfølgende brukeropplevelse

Figurene under viser noen andre modeller og deres bruksområde:



Informasjonssystem suksessmodell av DeLone og McLean brukes for å forstå suksessfull bruk av IS ved å identifisere, beskrive og forklare forholdet mellom seks kritiske dimensjoner: informasjonskvalitet, systemkvalitet, tjenestekvalitet, intensjon om bruk, brukertilfredshet og netto fordeler.



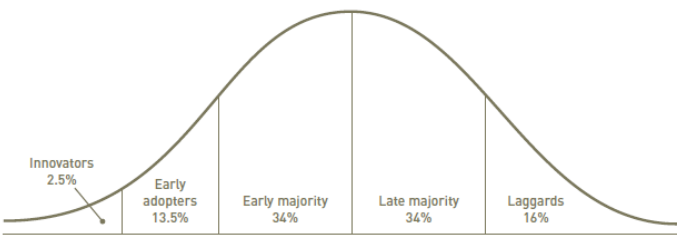
Task-Technology Fit brukes for å forstå hvordan teknologi kan påvirke ytelsen til et individ.

Viktig

Diffusjon av innovasjon teorien

Diffusjon av innovasjon teorien ble utviklet av E. M. Rogers for å forklare hvordan en ny ide eller produkt oppnår akseptanse og diffunderer gjennom en spesifikk populasjon eller del av en organisasjon. Denne teorien sier at tilpasningen til en innovasjon ikke vil skje med en gang for alle medlemmene i populasjonen, men er heller dratt ut over en periode. Noen medlemmer tilpasser seg raskt, mens andre bruker mer tid. **Teorien deler medlemmene inn**

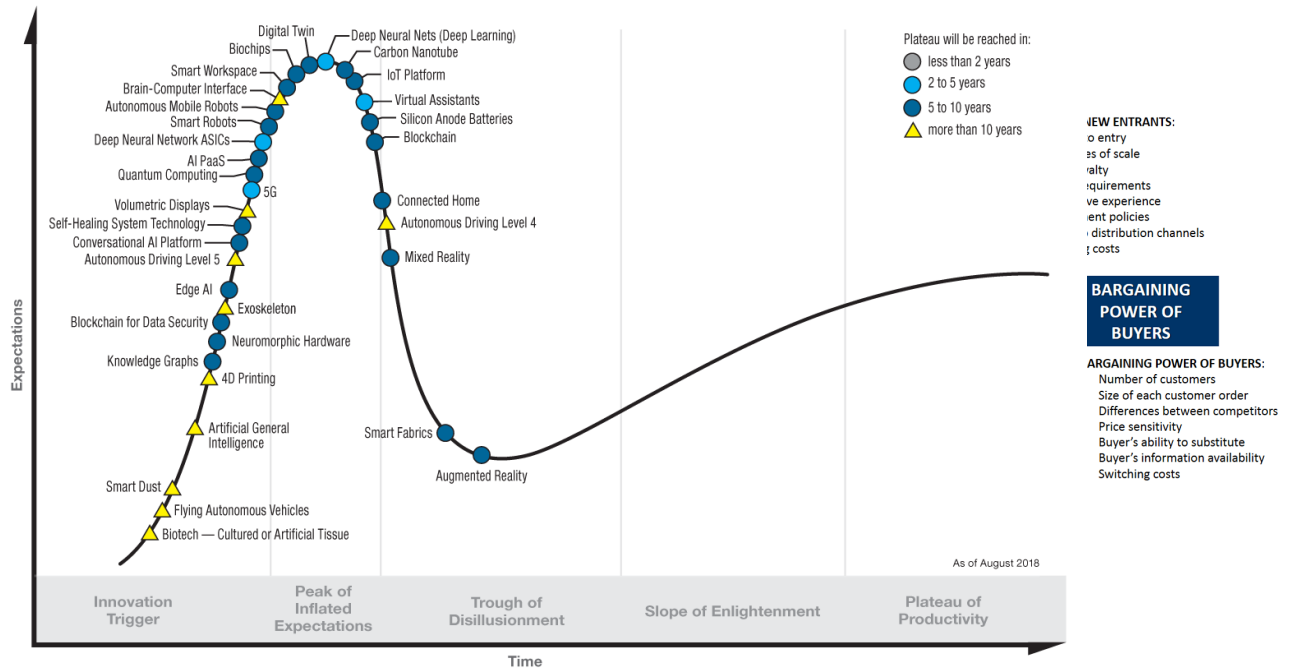
i fem kategorier som har ulik holdning mot tilpasningen (se figur). Når man skal introdusere en innovasjon er det viktig at man forstår egenskapene til kategoriene som vil hjelpe eller hindre tilpasningen av innovasjonen og deretter bruker passende strategi. Dette er viktig når man skal introdusere et nytt informasjonssystem.



Adopter Category	Characteristics	Strategy to Use
Innovator	Risk takers; always the first to try new products and ideas	Simply provide them with access to the new system and get out of their way
Early adopter	Opinion leaders whom others listen to and follow; aware of the need for change	Provide them assistance getting started
Early majority	Listen to and follow the opinion leaders	Provide them with evidence of the system's effectiveness and success stories
Late majority	Skeptical of change and new ideas	Provide them data on how many others have tried this and have used it successfully
Laggards	Very conservative and highly skeptical of change	Have their peers demonstrate how this change has helped them and bring pressure to bear from other adopters

Nye teknologier (F)

Figuren viser hype-syklus for nye teknologier som kom i 2018



Prosesen av å produsere en ny teknologi er som følger:

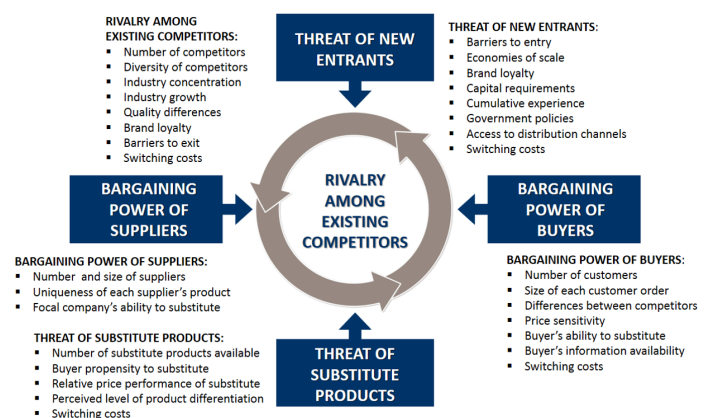
1. Ting sparkes i gang av et potensielt teknologisk gjennombrudd, der betydelig publisitet kommer av tidlige *proof-of-concept* historier og medieinteresse. Som regel eksisterer det ingen brukbare produkter og kommersiell levedyktighet er uprøvd.
2. Tidlig publisitet produserer en rekke suksesshistorier som ofte er ledsaget av feilscore. Noen selskaper tar grep, men de fleste gjør ikke det
3. Interessen avtar etterhvert som eksperimenter og implementasjoner ikke klarer å levere. Investeringen forsetter kun hvis de overlevende leverandørene forbedrer produktene og tilfredsstillir tidlige kunder
4. Det gis mer informasjon om hvordan teknologien kan være til nytte for bedrifter og det blir mer forståelse. Det lages andre- og tredjegerasjon produkter. Flere selskaper finansierer piloter, men konservative selskaper er fortsatt forsiktige
5. Mainstream-tilpasningen begynner å ta fart. Kriterier for å vurdere leverandørens levedyktighet er tydeligere definert. Hvis teknologien har mer enn et nisjemarked vil den fortsette å vokse.

Konkurransfordel

En konkurransfordel er en signifikant og ideelt sett langvarig fordel som et selskap har over sine konkurrenter. Det kan resultere i produkter med høyere kvalitet, bedre kundeservice og lavere kostnader. Det er en rekke faktorer som kan føre til at en organisasjon får konkurransfordel, for eksempel etablering av nisjemarked, reduserte kostnader og lavere priser.

Porters Five Forces modell

Porters Five Forces model beskriver fem av disse faktorene, og den kan brukes for å forstå konkurransen i organisasjonens miljø, slik at den kan identifisere den potensielle lønnsomheten ved en strategi. Når man forstår hvilke krefter i miljøet eller industrien



som kan påvirke lønnsomheten til organisasjonen, blir det enklere å tilpasse strategien deretter. De fem faktorene er:

1. **Rivalisering blant eksisterende konkurrenter** = industrier med høy konkurranse har ofte høye, faste kostnader for å entre eller forlate industrien, liten forskjell mellom produktene og mange konkurrenter. For å få konkurransefordel vil selskapene konstant analysere hvordan konkurrentene bruker ressurser og eiendeler.
2. **Trussel om nye deltagere** = det er lave kostnader for å entre eller forlate industrien og teknologien som trengs for å starte virksomheten er lett tilgjengelig. Dette resulterer i høy trussel om nye deltagere og et ønske om konkurransefordel for å «skremme bort» nye deltagere.
3. **Trussel med substitutt produkter og tjenester** = selskap som tilbyr en type goder eller tjenester blir truet av andre selskap som tilbyr lignende goder eller tjenester. I slike tilfeller vil bedrifter prøve å skaffe konkurransefordeler.
4. **Forhandlingskraft hos kunder** = store kunder kan påvirke selskap ved å true med å bytte til et annet selskap. Da vil selskapet øke konkurransefordelen for å holde på kundene
5. **Forhandlingskraft hos leverandører** = når leverandører har stor påvirkningskraft vil bedriften øke konkurransefordelen for å opprettholde deres forhandlingskraft. Leverandører kan også hjelpe en organisasjon til å få konkurransefordel

Miles og Snows organisasjonsstrategier (F)

Miles og Snows organisasjonsstrategier brukes for å sikre at alle handlinger som tas i en organisasjon jobber mot samme ønskede resultater. Dette gjøres ved å bestemme hvilken av fire unike strategier organisasjonen benytter seg av (se tabell). For eksempel er *prospector* (utbygger) en strategitype som er basert på innovasjon og vekst, der man fremmer risikotaking og søker etter nye muligheter og marked.

Strategy Type	Definition	Examples
Prospector	In innovative and growth oriented, searches for new markets and new growth opportunities, encourages risk taking	Amazon Google
Defender	Protects current markets, maintains stable growth, serves current customers	BIC Corporation YKK Group
Analyzer	Maintains current markets and current customer satisfaction with moderate emphasis on innovation	IBM Yahoo UBER
Reactor	No clear strategy, reacts to changes in the environments, drifts with events	NOKIA

Strategisk planlegging for konkurransefordel

For å være konkurransedyktig må bedriften være rask, fleksibel, innovativ, produktiv, økonomisk og kundeorientert. Den må justere IS strategien med generelle bedriftsstrategier og mål. Noen av strategiene som kan foreslås basert på Porters Five Force modell er:

- **Kostnadslederskap** – lever produkter og tjenester med lavest mulig kostnad
- **Differensiering** – lever ulike produkter og tjenester, slik at kunden får flere valgmuligheter eller høyere kvalitet
- **Nisjestrategi** – lever til et lite nisjemarked
- **Endre industristrukturen** – endre industrien slik at den passer bedre med organisasjonen
- **Lage nye produkter og tjenester** – periodisk introdusere nye produkter eller tjenester

Kapittel 7 – Elektronisk og mobil handel

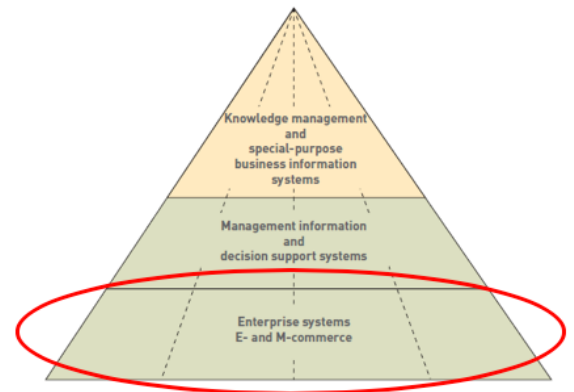
Introduksjon

Elektronisk og mobil handel er i stadig utvikling, og de gir nye måter å utføre virksomhet som innebærer både fordeler og ulemper.

Viktig

En introduksjon til E-commerce

Elektronisk handel (e-commerce) er utføring av business aktiviteter elektronisk over datanettverk. Disse aktivitetene kan være kjøp, salg, markedsføring, osv.. Business aktiviteter som er papirbaserte, tidkrevende og upraktiske for kunder blir ofte omdannet til e-commerce. Figuren viser **business informasjonssystem**, som ofte er integrert i ett produkt og kan leveres i samme software pakke. Som vi kan se er e-commerce en del av informasjonssystemet, sammen med enterprise system og m-commerce som vi skal se på senere.



Elektronisk handel involverer alle business transaksjoner som kan utføres elektronisk mellom selskap, forbrukere, offentlig sektor, osv. De **ulike typene e-commerce** er:

- **Business-to-Business (B2B)**
- **Business-to-Consumer (B2C)**
- **Consumer-to-Consumer (C2C)**
- **Consumer-to-Business (C2B)**
- **Government-to-Consumer (G2C)**
- **Government-to-Business (G2B)**
- **Government-to-Government (G2G)**



Business-to-Business e-commerce (B2B)

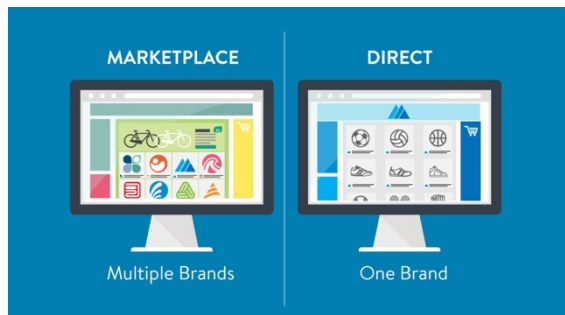
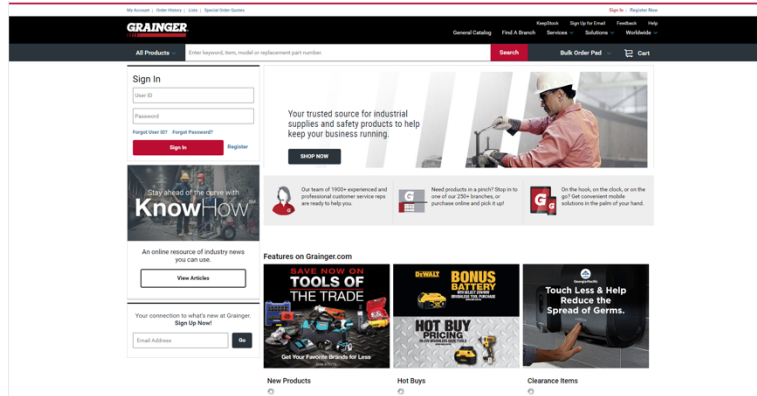
Business-to-Business (B2B) e-commerce er et subset av e-commerce, der alle deltagerne er organisasjoner. B2B er et nyttig verktøy som brukes til å koble sammen business partnere i en virtuell supply chain, slik at man kan kutte leveringstid og redusere kostnader. B2B markedet er svært stort og vokser hurtig. Viktige prioriteringer for B2B e-commerce inkluderer oppgradering av e-commerce plattformen, integrering av plattformen med regnskap og ordrehåndtering og etablering av mobile sider og apper. For at B2B handelen skal lykkes må kundene flyttes til nettet.

Mange organisasjoner vil bruke både:

- **Buy-side e-commerce** = handler goder og tjenester fra sine leverandører. Aktiviteter kan inkludere identifisering og sammenligning av konkurrerende leverandører og produkter, forhandling om priser og vilkår, bestilling, sporing, osv.
- **Sell-side e-commerce** = selger produkter til sine kunder. Aktiviteter kan inkludere muliggjøre kjøp på nettet, gi informasjon til kundene, fremme salg, osv.

Aktiviteter innenfor buy-side og sell-side e-commerce støtter value-chain til organisasjonen, siden de kan øke verdien til input. De kan også hjelpe organisasjonen med å gi lavere priser, bedre kundeservice, høyere kvalitet og mer unike produkter og tjenester.

Et eksempel på B2B e-commerce er Grainger, som er en B2B-distributør av produkter for vedlikehold, reparasjon og operasjoner. De har en mobil app som lar kundene raskt finne og bestille produkter med deres mobile enheter,



Som vi kan se på figuren kan B2B e-commerce deles inn i markeds plass der det selges produkter av flere merker, eller direkte der det selges produkter av ett merke.

Business-to-Consumer e-commerce

Business-to-Consumer (B2C) e-commerce er et subsett av e-commerce, der kunder er i direkte kontakt med en organisasjon uten mellommenn. Området Asia-Stillehavet står for over en tredjedel av verdens B2C salg, og det er forventet en hurtig vekst i Brasil, Russland, India og Mexico. **Årsaker til hurtig vekst i B2C e-commerce er:**

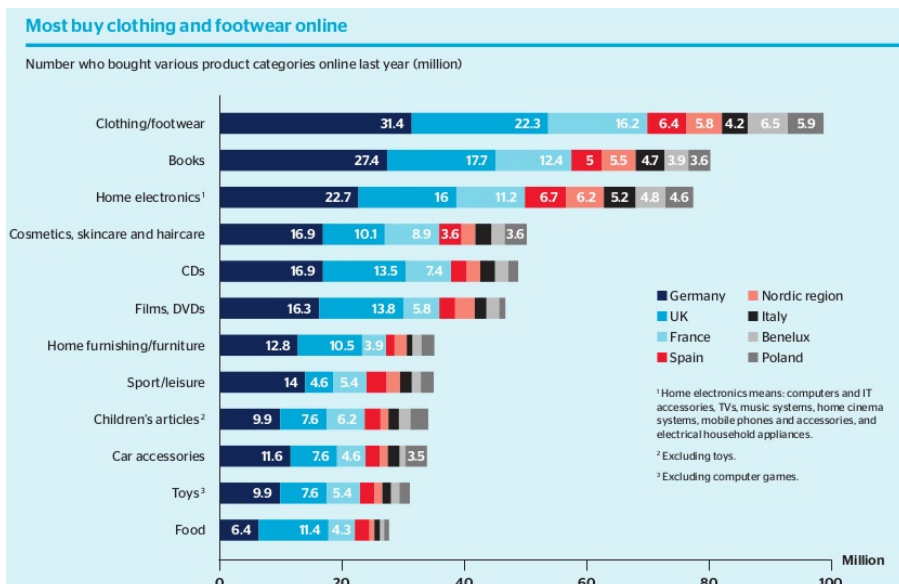
- Mange gode og tjenester er **billigere** på Internettet
- Online shoppere kan **designer personlige produkter**
- **Sosiale medier** brukes for å fremme produkter og nå kunder
- Internett lar kundene **sammenligne priser, egenskaper og verdi og sjekke andre kunders meninger** om produktet eller tjenesten

Tabellen viser mengde B2C e-commerce i ulike områder av verden.

Region	Sales (billions)					
	2012	2013	2014	2015	2016	2017
Asia-Pacific	\$301.2	\$383.9	\$525.2	\$681.2	\$855.7	\$1,052.9
North America	\$379.8	\$431.0	\$482.6	\$538.3	\$597.9	\$660.4
Western Europe	\$277.5	\$312.0	\$347.4	\$382.7	\$414.2	\$445.0
Central and Eastern Europe	\$41.5	\$49.5	\$58.0	\$64.4	\$68.9	\$73.1
Latin America	\$37.6	\$48.1	\$57.7	\$64.9	\$70.6	\$74.6
Middle East and Africa	\$20.6	\$27.0	\$33.8	\$39.6	\$45.5	\$51.4
Worldwide	\$1,058.2	\$1,251.4	\$1,504.6	\$1,771.0	\$2,052.7	\$2,357.4

Ved å bruke B2C e-commerce for å selge direkte til forbrukere, kan produsenter eller leverandører eliminere mellommenn mellom dere og forbrukerne, noe som kalles disintermediation. Dette vil i mange tilfeller redusere kostnader og gjøre supply chain mer effektiv, som igjen vil føre til større fortjeneste hos organisasjonen og lavere priser hos forbrukerne.

Som vi kan se på figuren til høyre er det vanligst at mennesker kjøper klær, bøker eller hjemmeelektronikk på nettet. Et eksempel på B2C e-commerce er Amazon, som selger en rekke ulike produkter via internasjonale nettsider over hele verden.



En stadig større trend er å undersøke produktet på nettet, for så å kjøpe det ved en lokal butikk. Salg i lokale butikker som skyldes online markedsføring og undersøkelse kalles web-påvirket salg. Loven sier at forhandlere på nettet ikke trenger å samle skatt i stater der de har ingen fysisk tilstedeværelse, noe som resulterer i at online shoppere ikke trenger å betale statlig eller lokal skatt på deres kjøp på nettet. Dette gir en fordel til forhandlere over nett sammenlignet med butikker som må samle skatt ved salg. Det resulterer også i tap av skatteinntekter for staten og lokale regjeringer.

Consumer-to-Consumer e-commerce

Consumer-to-Consumer e-commerce er et subsett av e-commerce, som involverer elektroniske transaksjoner mellom forbrukere tilrettelagt av en tredjepart. Et eksempel på C2C e-commerce er eBay, der forbrukere kan kjøpe og selge ting til hverandre gjennom siden.



Andre populære nettsider inkluderer Finn, airbnb, Etsy, Taobao, Craigslist, osv. C2C e-commerce har erstattet bruken av sider i aviser for å reklamere og selge personlige ting. **Selskap og individer som er involvert i C2C e-commerce må passe på at deres salg ikke bryter reglene i lovlige jurisdiksjoner hos ulike fylker, stater og land.** For eksempel kan land ha svært ulike lover angående salg av våpen.

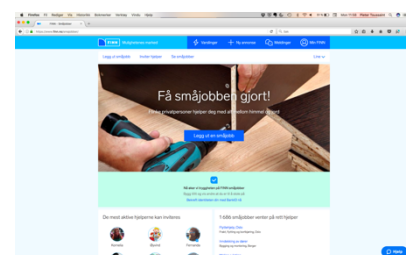


Tabellen under viser hovedforskjellen mellom B2B, B2C og C2C e-commerce:

Factors	B2B	B2C	C2C
Typical value of sale	Thousands or millions of dollars	Tens or hundreds of dollars	Tens of dollars
Length of sales process	Days to months	Days to weeks	Hours to days
Number of decision makers involved	Several people to a dozen or more	One or two	One or two
Uniformity of offer	Typically a uniform product offering	More customized product offering	Single product offering, one of a kind
Complexity of buying process	Extremely complex; much room for negotiation on quantity, quality, options and features, price, payment, and delivery options	Relatively simple; limited negotiation on price, payment, and delivery options	Relatively simple; limited negotiation on payment and delivery options; negotiations focus on price
Motivation for sale	Driven by a business decision or need	Driven by an individual consumer's need or emotion	Driven by an individual consumer's need or emotion

Consumer-to-Business (C2B) e-commerce

Consumer-to-Business (C2B) e-commerce er et subsett av e-commerce der forbrukere skaper verdi og organisasjonen forbruker denne verdien. For eksempel kan forbrukere gi omtaler eller nyttige ideer om nye produktutviklinger som organisasjonen tar som input. Forbrukere kan også tilby produkter og tjenester til organisasjoner, som dermed betaler forbrukeren. Et eksempel på dette er småjobber på Finn.no.



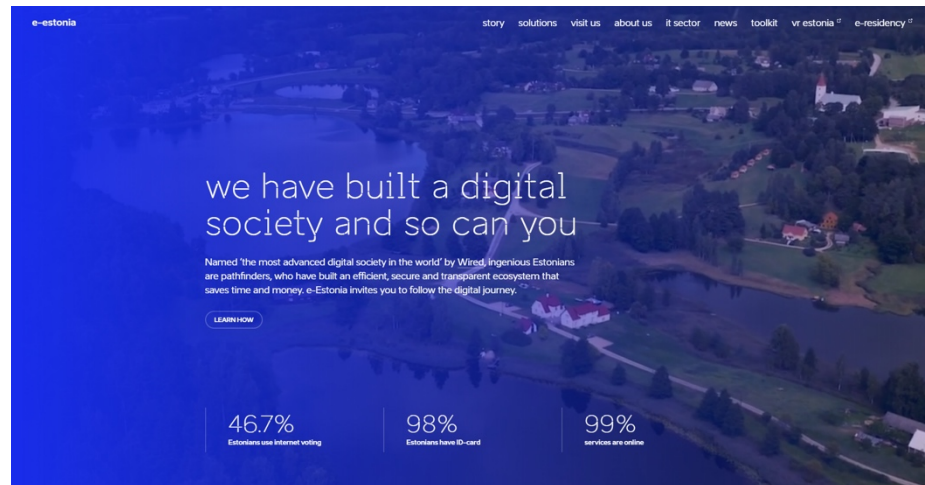
E-Government

E-Government er bruken av informasjon- og kommunikasjonsteknologi for å forenkle delingen av informasjon, øke hastigheten til prosesser som tidligere var papirbaserte og forbedre forholdet mellom innbyggere og myndighetene. Det finnes tre typer E-Government:

1. **Government-to-Citizen (G2C)** = brukes for å sende inn selvangivelser på nettet, fornye førerkortet, søke om studentlån, osv.
2. **Government-to-Business (G2B)** = lar myndighetenes anskaffelseskontorer kjøpe materialer og tjenester fra privat industri, lar bedrifter by på offentlige kontrakter og gjør det enklere for bedrifter å få informasjon om lover og reguleringer.

3. **Government-to-Government (G2G)** = støtter transaksjoner mellom organisasjoner i myndigheten. For eksempel kan det brukes for å utveksle informasjon om fødsel og død, arrestordre, mengde statlig støtte, osv.

Et eksempel er e-estonia som er en e-government i Estland, som innbyggerne kan bruke for å finne digital ID, stemme ved politisk valg, følge med på helsedata og skatt, osv. E-estonia er et informasjonssamfunn som samler digital data om innbyggerne i Estland. Dette har gjort at Estland er en av verdens mest utviklede digitale samfunn.



Viktig

En introduksjon til m-commerce

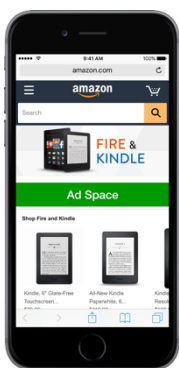
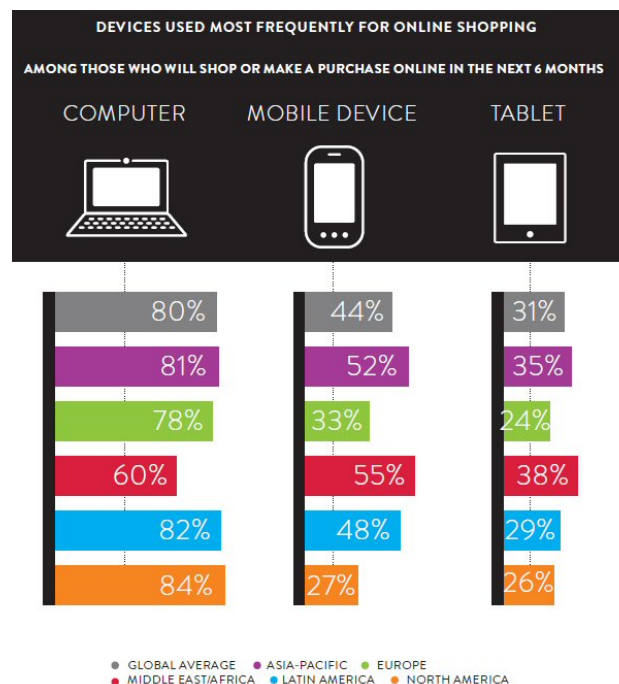
Mobil handel (m-commerce) innebærer bruken av mobile, trådløse enheter for å legge inn bestillinger og utføre business. Produsenter av mobile enheter (eks: Apple) samarbeider med kommunikasjonsbærere (eks: Telenor) for å utvikle trådløse enheter, relatert teknologi og tjenester som støtter m-commerce.

Mobil handel på verdensbasis

Markedet for m-commerce modner mye senere i Nord-Amerika sammenlignet med Vest-Europa og Japan. Dette skyldes at i Nord-Amerika er infrastrukturen til nettverket delt opp blant mange tilbydere, forbrukere betaler som regel med kredittkort og mange amerikanere kjenner ikke til mobile datatjenester. I Vest-Europa og Japan er det vanligere å kommunisere via trådløse enheter og forbrukerne er mer villige til å bruke m-commerce.

Det er forventet at m-commerce vil ha en hurtig vekst på verdensbasis, og i fjerde kvartal av 2018 utgjorde m-commerce 35 % av all e-commerce salg over hele verden. Merk: mobil handel er en form for e-commerce dersom man bruker en trådløs enhet for å handle på elektroniske nettverk. Det blir stadig flere mobile nettsider (eks: Amazon på figuren) som følge av bedre teknologier innen trådløs bredband, utviklingen av nyttige applikasjoner og produksjon av billige og kraftfulle mobile enheter.

Figuren til høyre viser hvilke enheter som er vanlig å bruke for online shopping i ulike deler av verden. M-commerce er mest vanlig i Afrika og Asia, noe som kan skyldes at mobile enheter er billigere enn datamaskiner.



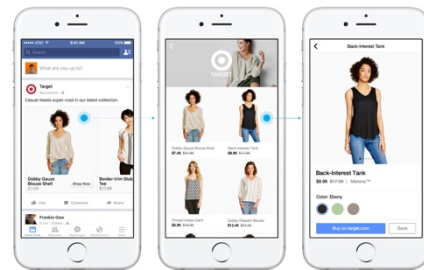
Rank	Company
1	eBay
2	Amazon
3	Apple
4	Burberry
5	John Lewis
6	Lush

Tabellen viser en rangering av m-commerce nettsider hos ulike organisasjoner.

Viktig

Sosial commerce (F)

Sosial commerce er et subsett av e-commerce som involverer sosiale medier, online media med sosial interaksjon og brukerbidrag, og det støtter kjøp og salg av produkter og tjenester på nettet. Det kan for



THE IMPORTANCE OF SOCIAL COMMERCE



87% of customers say that social media help them decide what to buy.



25% seek advice on social networks when choosing an apparel.



80% of followers try to reach out to brands via social media.

eksempel være en bruker som deler informasjon og råd om produkter, annonser som legges ut på Facebook eller produktomtaler som gis på Amazon. Sosial commerce har en viktig rolle i e-commerce, for eksempel oppgir 87 % av kunder at de bruker sosiale medier som hjelp til å bestemme hva de skal kjøpe.

56 customer reviews

★★★★★ 4.9 out of 5 stars



Fordeler ved elektronisk og mobil handel

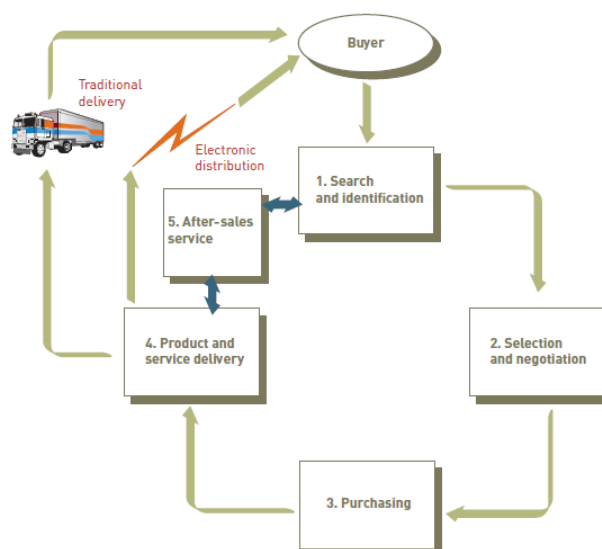
Fordeler ved bruk av elektronisk og mobil handel er:

- **Nå nye kunder** – etablering av e-commerce nettside lar bedriften nå nye kunder i det globale markedet. Lar også bedrifter fra utviklende land delta, slik at det hjelper til med å redusere gapet mellom rike og fattige land.
- **Redusere kostnader** – prosessen fra bestilling til levering kan involvere flere steg som krever mye tid og arbeid. Ved å redusere disse kan man fullføre flere salg innen samme tid og med økt nøyaktighet. Dermed kan bedriften redusere totale kostnader forbundet med produksjon, lagring og transportering. For eksempel kan man transportere flere produkter i én stor trailer istedenfor å dele det opp i mange små.
- **Øke hastigheten til flyten av goder og informasjon** – informasjon og goder kan flyte enklere og raskere fra selger til kjøper
- **Øker nøyaktigheten** – siden kjøperen selv skriver inn bestillingsinformasjon og produktspesifikasjon er det mindre sannsynlig at det skjer feil på leverandørens side.
- **Forbedrer kundeservice** – kundene kan bli mer fornøyde dersom de får mer detaljert informasjon om leveringsdato og status på bestillingen.

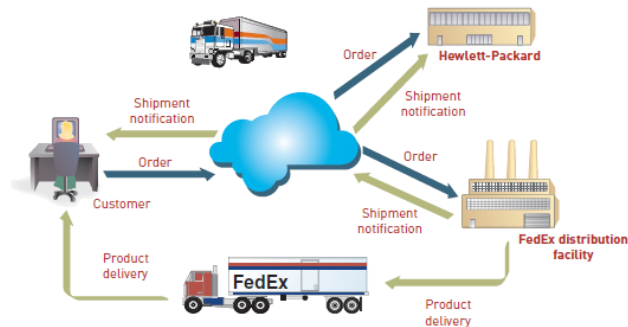
Multistage modell for e-commerce

For at e-commerce systemet skal være suksessfullt må det adressere stegene som forbrukeren oppfatter i livssyklusen til salget. De ulike stegene er:

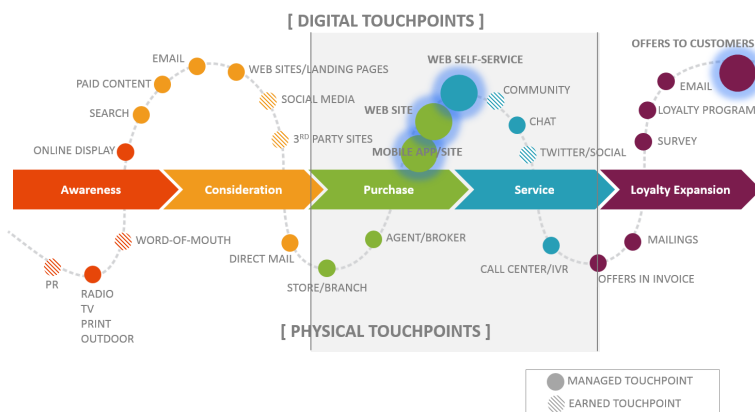
1. **Søk og identifisering** – på hjemmesiden til leverandøren kan kunden aksessere en produktkatalog og søke gjennom denne for å finne en bestemt enhet. Kunden kan oppgi antall enheter, og Web applikasjonen gir total kostnad og lar kunden velge type levering. Kunden kan besøke siden til andre leverandører.
2. **Seleksjon og forhandling** – kunden kan ha funnet enhetene ved ulike leverandører og undersøker prisen og leveringsmulighetene. Kvalitet og leverandørens tjenester kan også være viktig for å velge leverandør. B2B støtter forhandling mellom kjøper og selger, mens B2C vil ofte tilby produkter til en bestemt pris.
3. **Elektronisk kjøp av produkter og tjenester** – kunden fullfører bestillingen som gir de endelige avtalte vilkårene og prisen ved å sende et elektronisk skjema til leverandøren. Etablerte kunder kan ha satt opp kreditt hos leverandøren, mens nye kunder kan betale med kredittkort, faktura, osv. (krever større sikkerhetsjekk)



4. **Levering av produkter og tjenester** – produkter som programvare, musikk, bilder og videoer kan leveres raskt og billig over Internettet, mens andre produkter som klær, møbler og bøker kan leveres med trailere, vanlig mailtjeneste, osv. Mange produsenter og leverandører outsourcer denne prosessen med lagring, pakking, transportering og sporing til selskaper som DHL og FedEx. Disse har utviklet programvareverktøy og grensesnitt som kobler sammen kundens bestilling, produksjonen og inventarsystem med deres egne system med automatiserte varehus, kundesenter og transportnettverk. Målet er at overføringen av informasjon og inventar fra produsent til leveringselskap til forbruker skal være rask og enkel (se figur).



5. **Etter-salg tjenester** – informasjon om kunden blir lagret i kundedatabasen til leverandøren, og dette kan inkludere navn, adresse, telefonnummer, kreditthistorie og andre detaljer. Dette kan brukes for eksempel hvis kunden ikke mottar varen eller reklamerer på en vare.



Figuren til venstre viser en annen multistage modell for e-commerce som legger større vekt på skillet mellom de fysiske og digitale delene av de ulike stegene. For eksempel ser vi at selve kjøpet kan involvere digitale nettsider eller apper og fysiske butikker.

Utfordringer ved E-commerce

For å omdanne sine tradisjonelle business prosesser til e-commerce prosesser må bedriften overkomme flere utfordringer, spesielt for B2C e-commerce. Noen av disse er:

- **Håndtere personvernproblemer** = omtrent en tredjedel av alle voksne internettbrukere oppgir at de ikke vil kjøpe noe på nettet fordi de er bekymret for personvern. Det har vært flere situasjoner der organisasjoner har opplevd sikkerhetsbrudd hvor informasjon om kredittkort og personlig informasjon har blitt stjålet. I noen tilfeller kan dette føre til identitetstyveri. Det krever en substansiell investering å ivareta kundens privatliv og selskapet må være forberedt på dette. De fleste investerer i den seneste sikkerhetsteknologien og ansetter godt trent sikkerhetsekspert for å beskytte kundedataen.
- **Overkomme forbrukerens mangel på tillit** – mangel på tillit til online selgere er en av de vanligste grunnene til at forbrukere ikke kjøper ting på nettet. De er usikre på om selskapet eller personen de kjøper fra vil sende tingene de kjøper. For å overkomme denne utfordringen kan selskapet tilby lojalitetsprogram, dele feedback fra andre kunder, vise at bedriften har holdt på lenge, gjøre det klart at mye innsats har blitt lagt inn i nettsiden, reklamere i media, ordne kvalitetsstempel fra kjente organisasjoner, osv.
- **Overkomme globale problemer** = e-commerce og m-commerce gir enorme muligheter, siden det lar organisasjoner og enkeltindivider kjøpe og selge på et globalt marked. Samtidig innebærer disse mulighetene også en rekke utfordringer innenfor blant annet kultur, språk, tid, avstand, infrastruktur, valutakurs og lovverk.

For eksempel kan språkforskjeller gjøre det vanskelig å forstå innholdet på nettsiden, tidsforskjeller kan gjøre det vanskelig for noen kunder å komme i kontakt med kundeservice og nettsiden må ta hensyn til at kunder bruker en rekke ulike maskinvare- og programvareenheter.

Bruksområder for e- og m-commerce

E-commerce og m-commerce kan brukes på mange innovative måter for å forbedre operasjonene til en organisasjon. I likhet med andre typer teknologier vil de kun lykkes dersom de gir brukere virkelige fordeler. Vi skal nå se nærmere på noen bruksområder for elektronisk og mobil handel.

Små- og storhandel

E-commerce blir brukt mye innenfor både små- og storhandel. Elektronisk salg, ofte kalt e-salg, er når bedrifter selger produkter eller tjenester direkte til forbrukere via elektroniske butikkfronter. Disse frontene har ofte digitale kataloger og handlevogner. En viktig del av den elektroniske storhandelen er goder og tjenester for MRO (*Manufacturing, Repair og Operations*), som kan være kontorforstyringer, motorer, verktøy, osv. Ofte vil slike varer utgjøre en stor del av innkjøpene til et selskap, og mye tid kan sløses bort på dårlige kjøpssystem eller situasjoner der utstyr mangler. En e-commerce programvare kan brukes for å sammenligne funksjonelt like varer, kombinere kjøp for å redusere kostnader, redusere tidsbruken, osv.

Produksjon

Ved å flytte operasjoner i supply chain til Internett kan produsenter øke fortjeneste og forbedre kundeservicen. For eksempel kan de lage en elektronisk utveksling, som er et forum der produsenter, leverandører og konkurrenter kjøper og selger goder, deler markedsinformasjon og kjører operasjoner slik som lagerkontroll. Dette forumet gir blant annet raskere bevegelse av råmaterialer og ferdige produkter.

Markedsføring (F)

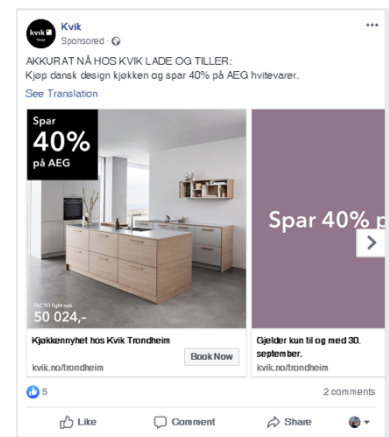
Internett har en rekke egenskaper som gjør at bedrifter kan samle informasjon om kundenes oppførsel og preferanser, slik at selskapet får et nærmere forhold til deres kunder. Internettannonser bruker denne dataen til **markedssegmentering**, der det identifiseres spesifikke markeder som deretter utsettes for skreddersydde annonseringsmeldinger. Potensielle kunder blir som regel delt inn i undergrupper basert på demografiske egenskaper som alder, kjønn, inntekt, osv. E-commerce gjør dermed at markedsføringen kan bli mer personlig, siden den kan baseres på søkehistorikken og interesser hos kunden.

Viktig

Reklame

Siden m-commerce som regel har kun én bruker er de ideelle for å aksessere personlig informasjon og sende målrettede meldinger til bestemte forbrukere. Mobil reklame kan plasseres på blant annet nettsider og apper, og mobile annonsevisninger blir ofte kjøpt som *cost per thousand (CPM)*, *cost per click (CPC)* eller *cost per action (CPA)*. Annonseren betaler dermed kun hvis kunden klikker seg inn på reklamen. Suksess kan måles som antall brukere som nås, *click through rate (CTR)* eller antall handlinger brukerne utfører (eks: antall

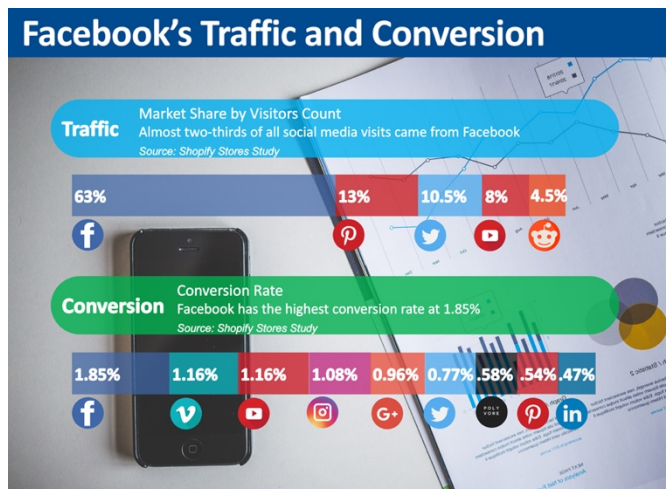
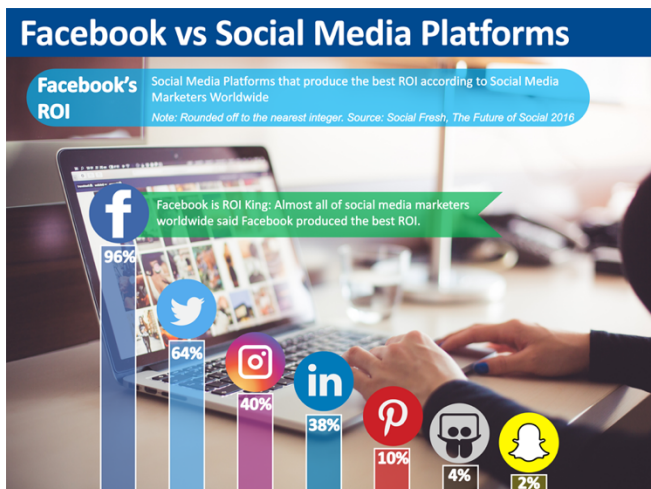
Viktig



nedlastninger). Denne dataen kan brukes for å måle hvor effektiv reklamen er og selskap vil derfor ofte betale ekstra for å motta den. Nettverk for mobil reklame kan deles inn i blind, premium blind og premium nettverk, men det er ingen klar skille mellom disse. Tabellen viser egenskaper ved de tre typene.

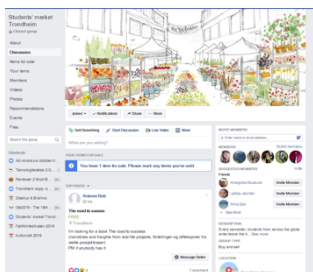
Characteristic	Blind Networks	Premium Blind Networks	Premium Networks
Degree to which advertisers can specify where ads are run	An advertiser can specify country and content channel (e.g., news, sports, or entertainment) on which the ad will run but not a specific Web site.	Most advertising is blind, but for an additional charge, the advertiser can buy a specific spot on a Web site of its choice.	Big brand advertisers can secure elite locations on top-tier destinations.
Predominant pricing model and typical rate	CPC (e.g., \$0.01 per click)	CPM (e.g., \$20 per thousand impressions)	CPM (e.g., \$40 per thousand impressions)
Examples	Admoda/Adultmoda AdMob BuzzCity InMobi	JumpTap Madhouse Millennial Media Quattro Wireless	Advertising.com/AOL Hands Microsoft Mobile Advertising Nokia Interactive Advertising Pudding Media YOC Group

Figurene under viser at Facebook er den sosiale medieplattformen som gir høyest ROI (Return Of Investment), altså gir mest igjen til virksomheten for hver brukte markedsføringskrone. Dvs. Facebook genererer mest klikk på reklamene.



Byttehandel (F)

Etter den nylige økonomiske nedgangen har mange mennesker og bedrifter begynt med byttehandel for å skaffe seg goder og tjenester. Tabellen viser noen nettsider som støtter denne aktiviteten. Bedrifter bruker byttehandel for å redusere overflødig inventar, få nye kunder og unngå å betale kontanter for råmaterialer eller tjenester, mens kunder bruker byttehandel istedenfor å betale med kontanter. Et annet eksempel er Marketplace på Facebook (figur til venstre).



Web site	Purpose
Craigslist.org	Includes a section where users can request an item in exchange for services or exchange services for services
Swapagift.com	Enables users to buy, sell, or swap merchant gift cards
Swapstyle.com	Allows users to swap, sell, or buy direct women's accessories, clothes, cosmetics, and shoes
Swaptrees.com	Users trade books, CDs, DVDs, and video games on a one-for-one basis
TradeAway.com	Enables users to exchange a wide variety of new or used items, services, or real estate

Retargeting

Omtrent 74 % av alle handlevogner på nettet blir forlatt, noe som representerte en verdi på 4 trillioner dollar i 2013. Retargeting er teknikken som brukes for å få tilbake disse kjøperne ved å bruke målrettet og personlige reklamer som retter kjøperne tilbake til nettsiden. For eksempel kan det vise klærne som kunden hadde valgt ut og evt. kommentarer fra andre kunder som kjøpte samme klær. **Retargeting sikrer at potensielle kunder ser relevante og målrettede reklamer for produkter som de allerede har vist interesse for.**

Andre bruksområder

Andre bruksområder for e- og m-commerce er:

- **Prissammenligning** – det lages mobile apper som lar kundene sammenligne priser og produkter på nettet. For eksempel kan RedLaser brukes for å sammenligne priser ved å skanne barkoden til produktet.
- **Kuponger** – mange bedrifter tilbyr digitale kuponger som sendes til kunden via apper, nettsider, mail eller meldinger. Nylig har store forhandler begynt med

proximity marketing, der det brukes Bluetooth-sendere for å levere kuponger til kunder som allerede er i butikken. Senderne kan plasseres på hyller eller produkter

- **Investering og finans** – internett har revolusjonert industrien for finans og investering, som tidligere hadde mange ineffektive områder. Online handel lar investorer nøye undersøke og kjøpe aksjer på noen få sekunder med lite ekstra kostnader.
- **Bank** – kunder kan bruke nettjenester til banken for å sjekke balansen på sparekontoer, overføre penger mellom kontoer og betale regninger. Mange banker tilbyr også mobile apper for å utføre en rekke tjenester.

Strategier for suksessfull e- og m-commerce (ikke F)

For at en organisasjon skal være suksessfull i elektronisk og mobil handel må den definere og utføre en effektiv strategi. En effektiv nettside må være enkel å bruke og oppnår målene til selskapet, samtidig som den er trygg, sikker og billig å sette opp og vedlikeholde.

Definere en effektiv strategi

De fleste e-commerce modellene inkluderer tre grunnleggende komponenter:

1. **Fellesskap** – et lojalt samfunn av kunder kan bygges opp av diskusjonsforum og andre sosiale handleverktøy. Disse kundene er engasjerte i selskapet og dets produkter og tjenester.
2. **Innhold** – tidsriktig, nyttig og nøyaktig innhold sørger for at kunder returnerer til nettsiden
3. **Handel** – kunder og bedrifter betaler for å kjøpe goder, informasjon eller tjenester

Definere, etablere, bygge og vedlikeholde nettsiden

Når en organisasjon skal bygge en nettsiden, må den først bestemme hvilke oppgaver nettsiden skal utføre. For eksempel kan det være å gi generell informasjon om organisasjonen eller produkter og tjenester organisasjonen selger. Disse oppgavene kan endre seg etterhvert som nettsiden får flere brukere eller organisasjonen får nye mål. Organisasjonen må deretter bestemme om de vil bygge og operere nettsiden selv eller outsource dette til et eksterne firma. Internett har enorme mengder e-commerce nettsider, så det er viktig at organisasjonen sørger for at kunder finner deres nettside. For å skape trafikk på nettsiden kan de registrere et domenenavn som er relevant til bedriften, for eksempel www.soccerstuff4u.com. De kan også gjøre nettsiden mer søkemotorvennlig ved å ofte legge til nytt innhold på nettsiden, få andre populære sider til å dele linken til nettsiden, inkludere metatagger på hjemmesiden, osv.

Operatører for nettsiden må konstant følge med trafikken og responstiden som besøkende opplever. Dersom kunder opplever ulogiske forsinkelser mens de forsøker å se på eller bestille produkter eller tjenester, vil de som regel forlate nettsiden og heller besøke nettsiden til en konkurrent. For å sikre akseptabel responstid ved økt trafikk, kan det være nødvendig å endre programvare, databaser og maskinvare som nettsiden kjører på. Operatører må også følge med på nye trender og utviklinger innenfor e-commerce, slik at de er forberedt på å utnytte nye muligheter. For eksempel har det blitt populært med personlige nettsider, der nettsidene skreddersys får å rette seg mot individuelle kunder.

Teknologisk infrastruktur (ikke F)

Suksessfull implementering av e-commerce krever signifikante endringer i eksisterende business prosesser og investering i IS teknologi. Denne teknologien må kunne støtte store mengder transaksjoner med kunder, leverandører og andre business partnere rundt om i verden. Noen komponenter i infrastrukturen til teknologien er maskinvare for webserver og programvare for webserver og e-commerce (katalog, handlevogn, osv.).

Elektronisk betaling (F)

En annen viktig komponent ved infrastrukturen til e-commerce er et system for elektronisk betaling. E-commerce teknologi benytter seg av brukeridentifisering og kryptering for å beskytte transaksjoner. Betalingen kan gjøres på flere ulike måter, for eksempel kredittkort, smart betaling, elektronisk kontanter, osv. Jo flere betalingsmåter nettsiden tilbyr, desto større er mengden kjøp som gjennomføres på nettsiden. Det har også blitt mulig å betale ved å bruke mobiltelefonen som et kredittkort.

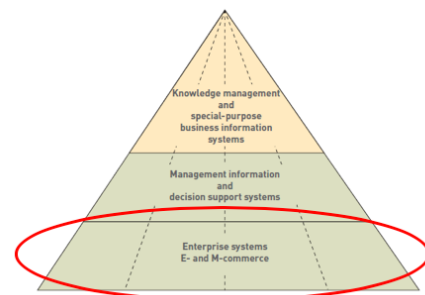


Klarna.



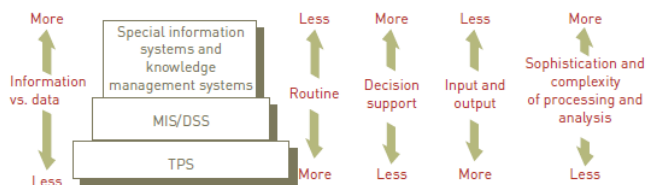
Kapittel 8 – Enterprise system

Enterprise system sørger for at data kan deles på tvers av alle business funksjoner og alle nivåer med management, slik at det støtter beslutningstaking som trengs for å drive organisasjonen. Vi skal se på utfordringer og fordeler med disse systemene, men begynner med å introdusere Transaksjon prosessering system.



Transaksjon prosessering system (TPS)

Transaksjons prosessering system (TPS) brukes for å fange og prosessere grunnleggende data om hendelser som påvirker organisasjonen (transaksjoner), og det bruker denne dataen for å oppdatere offisielle register i organisasjonen. Dette inkluderer oppføring av bestillinger, kontroll av varelager, lønningssystem, leverandørgjeld, kundefordringer, osv. Input til disse systemene er grunnleggende transaksjoner, som for eksempel kundebestillinger eller timeregister. Prosesseringen involverer samling, korreksjon, behandling og lagring av data og produksjon av dokumentasjon. Resultatet er at registrene til organisasjonen blir oppdatert, slik at de representerer statusen til operasjonen ved siste prosesserte transaksjon. Et eksempel på en TPS er et automatiseringssystem som mottar kundebestillinger fra mail, fax, telefon og e-commerce nettside og håndterer dette automatisk. Dette kutter prosesseringstiden og øker nøyaktigheten. Et annet eksempel er en nettbasert HR plattform som kan behandle lønn og andre fordeler, slik at selskapet kan øke antall ansatte uten å øke mengde HR personale.



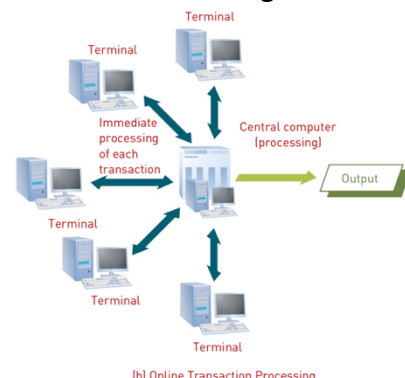
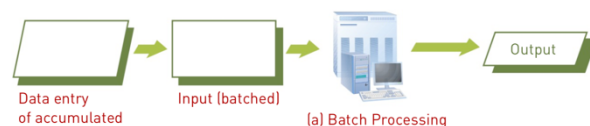
En TPS bruker store mengder input og output data for å oppdatere selskapets register om bestillinger, salg, kunder, osv. TPS gir likevel liten støtte for selve beslutningstakingen. **I stedet kan resultatet til en TPS brukes som verdifull input til management**

informasjonssystem (MIS), beslutningstaking støttesystem (DSS) og kunnskap management system (KMS).

Tradisjonell transaksjon prosesseringsmetoder

TPS bruker to ulike metoder:

- **Batch prosesseringssystem = prosessering av data der business transaksjoner blir akkumulert over en periode og prosessert som en gruppe (batch).** Lengden til perioden avhenger av brukerens behov, for eksempel kan kundebetalingen behandles på daglig basis. **Den essensielle egenskapen ved batch prosesseringssystem er forsinkelsen mellom en hendelse og prosesseringen av den relaterte transaksjonen som oppdaterer registrene til organisasjonen.** Batch prosessering er kostnadseffektivt, og det brukes ofte for lønning og fakturering.
- **Online transaksjon prosessering (OLTP) = prosessering av data der hver transaksjon blir prosessert med en gang de skjer, uten forsinkelsen forbundet med akkumulering av transaksjoner til en batch. Dette fører til at dataen i et online system til enhver tid reflekterer nåværende status.** Denne typen prosessering brukes av selskap som krever tilgang til nåværende data, som for eksempel flyselskap og aksjeselskaper. For mange organisasjoner vil OLTP tilby raskere og mer effektiv tjeneste, noe som kan gi dem verdi i kundens perspektiv.

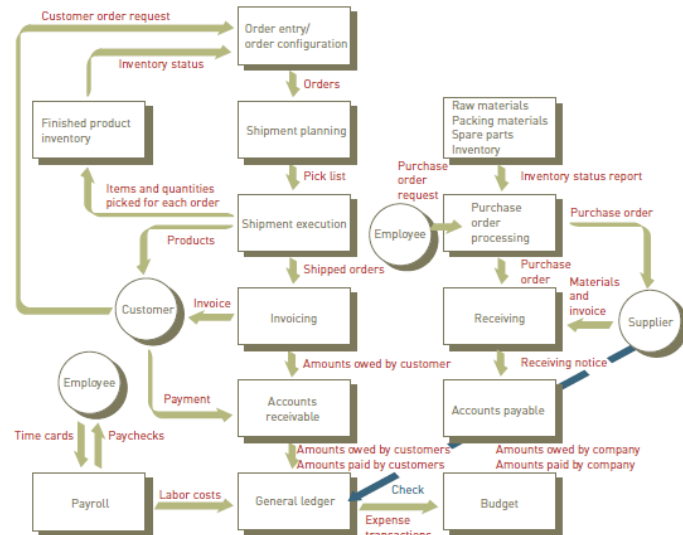


Hvilken transaksjon prosesseringsmetode som passer best for en organisasjon, avhenger av behovene og målene til organisasjonen. Behovet for nåværende data innenfor beslutningstaking, gjør at mange organisasjoner går fra batch prosesseringsystem til online transaksjon prosessering. Organisasjoner forventer at TPSs skal sikre at dataen som behandles er nøyaktig og fullstendig, de skal fange, prosessere og oppdatere databaser med business data som trengs for å støtte rutineaktiviteter, de skal unngå å behandle falske transaksjoner, de skal produsere tidsriktige responser og rapporter, de skal forbedre kundeservicen og de skal hjelpe bedriften med å oppnå en konkurransefordel.

Figuren viser flyten av viktig informasjon fra en TPS til en annen ved en produksjonsorganisasjon. Når transaksjoner som entrer et system blir prosessert, lages det nye transaksjoner som flyter inn i et annet system.

En TPS vil vanligvis inkludere følgende system:

- **Ordre prosesseringsystem** – kvitteringen til en kundebestilling vil begynne prosesseringsflyten, som deretter sjekker om det er tilstrekkelig inventar for å fyller bestillingen. Hvis det er tilfellet, vil kundeleveringen planlegges slik at den tilfredsstillir kundens ønskede mottaksdato. På dagen bestillingen skal sendes blir det skrevet ut en *pick list* ved varehuset. Ansatte samler tingene som trengs for å fyller bestillingen og inventaret blir oppdatert. Når bestillingen er fullført og sendt, blir det laget en kundefaktura (*invoice*) og en kopi blir inkludert i forsendelsen til kunden.
- **Regnskapssystem** – sporer flyten av data som er relatert til alle kontantstrømmer som påvirker organisasjonen. For eksempel vil kundefakturaen også sendes til kundefordringsystemet (*Accounts receivable*) for å oppdatere kontoen til kunden. Når kunden har betalt fakturaen vil betalingsinformasjonen brukes for å oppdatere kontoen på nytt. De nødvendige regnskapstransaksjonene sendes til *General ledger* systemet som sporer mengden kunden skylder og mengden som skyldes til leverandøren. Data om hva kunden skylder og har betalt til selskapet og hva selskapet har betalt leverandører og andre sendes til *General ledger* systemet som registrerer og rapporterer alle finansielle transaksjoner i selskapet.
- **Kjøpssystem** – støtter forretningsfunksjonen for innkjøp, og det inkluderer varelagerkontroll, prosessering av innkjøpsordre, mottak og leverandørgjeld. Når ansatte har identifisert en mangel i varelageret, vil de lage en innkjøpsordre. Informasjon om innkjøpsordren flyter til mottakssystemet og leverandøren. Leverandøren sender varen og faktura som matches mot original bestilling og mottatt rapport. Hvis dataen er fullstendig og komplett lages det en sjekk.

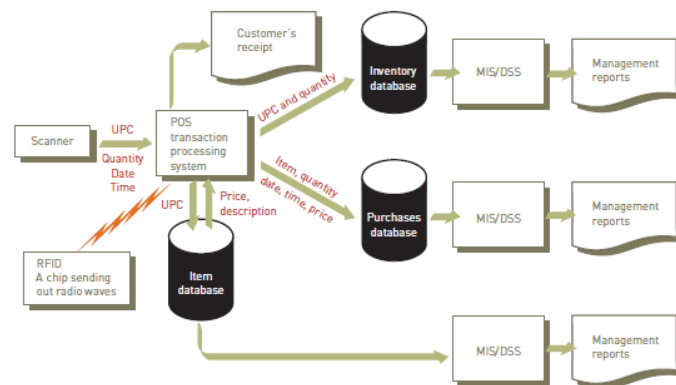


Mange organisasjoner implementerer et integrert sett med TPSs som håndterer de fleste eller alle aktivitetene beskrevet over og vist på figuren. Dataen flyter automatisk fra en applikasjon til en annen uten forsinkelser eller behov for å legge inn data på nytt. Fordeler ved å bruke TPS er at de reduserer kostnader (mindre arbeidstimer), gir bedre informasjonssamling, reduserer overflødig inventar, gir bedre kundeservice (sporing), osv. **En TPS kan dermed hjelpe organisasjonen i å nå deres mål om å redusere kostnader og øke produktiviteten, effektiviteten, kvaliteten og kundetilfredsheten.**

Transaksjon prosessering aktiviteter

Alle TPSs utfører et felles sett med grunnleggende aktiviteter. En TPS fanger og prosesserer data som beskriver grunnleggende business transaksjoner. Denne dataen brukes for å oppdatere databaser og produsere en rekke rapporter. Dataen går gjennom en **transaksjon prosesseringssyklus** (se figur), som involverer følgende aktiviteter:

1. **Data samling** – samling av all data som er nødvendig for å fullføre transaksjonsprosesseringene. Det begynner med en transaksjon (eks: kundebestilling) og resulterer i data som blir input til TPS. Dataen bør fanges ved kilden og registreres nøyaktig og tidsriktig, med liten manuell innsats. Dataen bør være elektronisk eller digital, slik at den kan legges direkte til datamaskinen. Dette kalles **kildedata automatisering** (*source data automation*), og et eksempel er kasseprosessen på butikker, der UPC koder leses av en skanner istedenfor at en ansatt må skrive inn koden manuelt. Denne avlesningen gir input til *point of sale* (POS) TPS som regner ut totalen kunden skal betale og oppdaterer inventar- og kjøpsdatabasen til butikken (se figur). Resultatet fra TPS brukes i MIS eller DSS for å lage manager rapporter som brukes ved bestilling av varer og i salgsanalyser.
2. **Data redigering** – det sjekkes om dataen er gyldig og fullstendig for å detektere problemer. For eksempel må data om kvantitet og kostnader være numerisk.
3. **Data korrigering** – ugyldig data fører til at systemet utløser feilmeldinger som gir problemet, slik at det blir enklere å korrigere dataen. Korrigeringen går ut på å legge inn dataen på nytt, siden den ikke ble skrevet inn eller skannet riktig første gangen. For eksempel hvis UPC koden blir lest feil eller ikke eksisterer i tabellen over gyldige UPC, blir den ansatte bedt om å reskanne enheten eller manuelt skrive inn koden.
4. **Data prosessering** – det utføres beregninger og andre datatransformasjoner relatert til business transaksjoner. Dette kan inkludere klassifisering av data, sortering av data inn i kategorier, osv. For eksempel gange antall timer med timelønn i lønssystem TPS.
5. **Data lagring** – en eller flere databaser blir oppdatert med nye transaksjoner. Denne dataen kan videre brukes av andre systemer (eks: MIS) for rapportering og beslutnings-taking. Transaksjonsdatabasen kan derfor ha stor påvirkning på organisasjonen.
6. **Dokument produksjon** – det lages output register, dokumenter og rapporter. Et eksempel er lønnsjekker. Resultat fra en TPS vil ofte brukes som input til andre system.



Enterprise system

Viktig

En organisasjon bruker et enterprise system for å støtte driften og styringen av virksomheten, siden systemet sikrer at informasjon kan deles blant autoriserte brukere på tvers av alle virksomhetsfunksjoner og alle nivåer for management. Dersom organisasjonen bruker flere transaksjon prosessering system (TPSs) som hver støtter kun én virksomhetsfunksjon eller én avdeling i organisasjonen, kan dette føre til manglende eller inkonsistent informasjon. For å unngå dette problemet, kan organisasjonen i stedet bruke et enterprise system som benytter seg av en database med viktig data om drift og planlegging som kan deles av alle. Bedrifter bruker enterprise system for å utføre mange av deres daglige aktiviteter innenfor produktforsyning, distribuering, salg, markedsføring, HR, produksjon, osv. Uten disse systemene ville registrering og prosessering av business transaksjoner brukt enorme mengder av ressursene til organisasjonen. Målet til enterprise system er å tilfredsstille kundene og gi signifikante fordeler ved å redusere kostnader og forbedre tjenester.

Obs: denne definisjonen varierer. Noen sier at SCM, CRM og product lifecycle management er deler i en ERP, mens andre sier at det er enterprise system

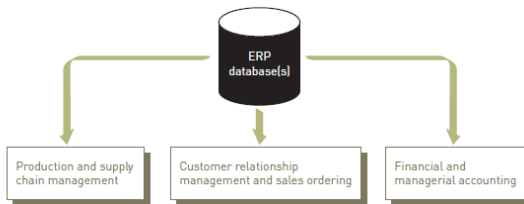
Tre viktige typer enterprise system, som vi nå skal se nærmere på, er:

- **Enterprise Resource Planning (ERP)**
- **Supply Chain Management (SCM)**
- **Customer Relationship Management (CRM)**

Viktig

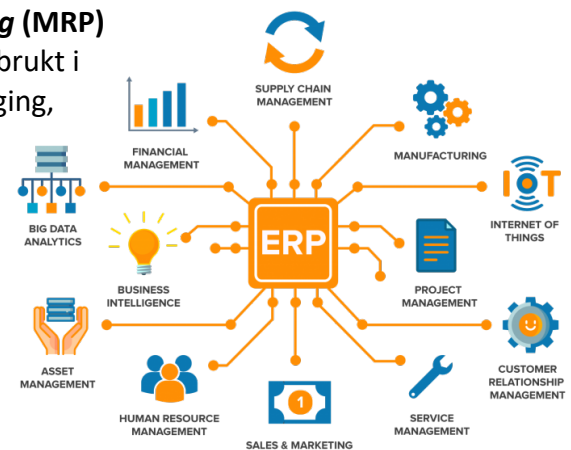
Enterprise Resource Planning (ERP)

Enterprise Resource Planning (ERP) er et sett med integrerte program som styrer bedriftens essensielle business prosesser for en hel organisasjon. En business prosess er et sett med koordinerte og relaterte aktiviteter som tar en eller flere typer input og lager en output som har verdi for en kunde av denne prosessen. Kunden kan være ekstern (eks: kjøper goder fra selskapet) eller intern (eks: ansatt i annen avdeling). Et eksempel på en business prosess er å fange en salgsordre, der input er en kundebestilling og output er bestillingen. **Kjernen ved ERP er en database som deles av alle brukere**, slik at alle business funksjoner har tilgang til tidsriktig og konsistent data, som dermed kan brukes i beslutningstaking og planlegging (se figur).



ERP systemer utviklet seg fra *Material requirement planning (MRP)* systemer som ble laget på 1970-tallet.

Disse systemene ble brukt i produksjonsselskap for å koble sammen produksjonsplanlegging, inventarkontroll og kjøpsfunksjoner. På slutten av 1980-tallet observerte mange organisasjoner at deres TPS manglet integreringen som trengs for å koordinere aktiviteter og dele verdifull informasjon på tvers av alle business funksjoner i bedriften. Dette førte til større kostnader og dårligere kundeservice. Derfor begynte den utfordrende implementeringen av ERP, og **store organisasjoner var de første til å bruke ERP.**



Viktig

Fordeler og ulemper ved ERP

ERP gir integrert programvare som tilfredsstiller behovet for bred tilgang til sanntidsinformasjon, som skyldes økt global konkurranse, større behov for kontroll over total kostnad og produktflyt og mer kundeinteraksjon. Fordelene ved å bruke ERP er:

1. **Bedre tilgang til kvalitetsdata for beslutningstaking** – ERP systemer bruker en integrert database der ett datasett støtter alle business funksjoner. Dataen er integrert fra starten av, slik at man slipper å samle data fra flere funksjoner eller forene data fra flere applikasjoner. Organisasjonen blir sømløs og det legges til rette for beslutningstaking, bedre kundeservice og sterkere forhold til leverandører og kunder.
2. **Eliminering av kostbare, lite fleksible system** – ERP gjør at organisasjoner som bruker opptil hundrevis av separate system, kan erstatte disse med ett enkelt, integrert sett med applikasjoner. Disse systemene kan være flere tiår gamle og vanskelige å reparere eller oppdatere som følge av dårlig dokumentasjon. Slike system kan hindre organisasjonen fra å utvikle seg og forbli konkurransedyktig. ERP system gjør at evnene til organisasjonens IS system kan matche deres forretningsbehov som stadig utvikles.
3. **Bedre arbeidsprosesser** – høy konkurranse gjør at bedrifter må ha business prosesser som er strukturerte, effektive og kundeorienterte. ERP system inkluderer applikasjoner som støtter «beste praksis», som er de mest effektive måtene å utføre business prosesser på, funnet ved å observere ledende organisasjoner i samme industri og funn fra forskning og konsulenter. Implementasjon av ERP system vil derfor sikre at arbeidsprosessene er basert på de beste praksisene i industrien.

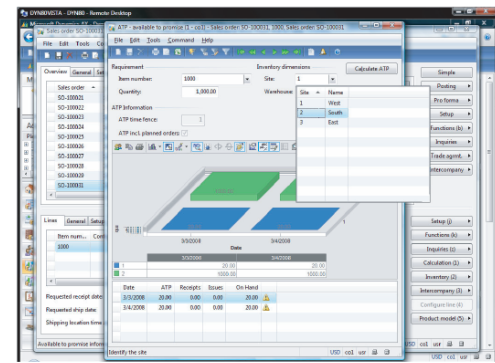
4. **Mulighet til å oppgradere og standardisere den teknologiske infrastrukturen** – når organisasjoner implementerer et ERP system, får de muligheten til å oppgradere informasjonsteknologien de bruker, for eksempel maskinvare, operativsystem og databaser. Ved å lage en standard med færre teknologier og leverandører kan organisasjonen redusere kostnader forbundet med vedlikehold, støtte og opplæring.

Ulempene assosiert med ERP system er at implementeringen er tidskrevende, vanskelig og kostbar. Det kan også være utfordrende å integrere ERP systemet med andre system.

Ulike typer ERP-leverandører

ERP system brukes av ulike typer organisasjoner (eks: produksjonsselskap, universitet, butikker, osv.) som har forskjellige behov. I tillegg vil behovene til en stor, multinasjonal organisasjon være forskjellig fra behovene til en liten, lokal organisasjon. Én ERP programvareløsning vil derfor ikke passe alle organisasjoner. **Leverandører av ERP er delt inn i tre klasser, avhengig av hvilken kundetype de er rettet mot:**

1. **Tier I leverandør** – retter seg mot store, multinasjonale bedrifter med flere geografiske lokasjoner og årlig inntekt som overgår 1 billion dollar. Tier I løsninger er svært komplekse og krever kostbar implementering og støtte. Eksempler er Oracle og SAP.
2. **Tier II leverandør** – retter seg mot medium-store selskap med en eller flere geografiske lokasjoner og årlig inntekt mellom 50 millioner og 1 billion dollar. Tier II løsninger er mindre komplekse og mindre kostbare å implementere og støtte. Eksempler er Oracle, SAP, Microsoft og Epicor
3. **Tier III leverandør** – retter seg mot mindre selskap som ofte har én geografisk lokasjon og årlig inntekt mellom 10 millioner og 50 millioner dollar. Tier III løsninger er relativt enkle og lite kostbare å implementere og støtte. Eksempler er Microsoft, Abas, PDS, Plex, Syspro, osv.



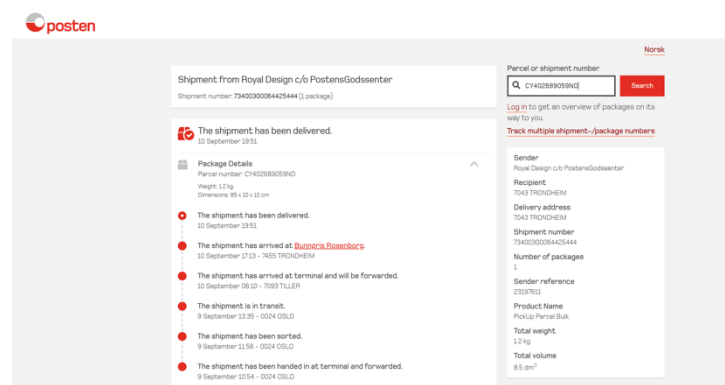
Microsoft Dynamics ERP programvare

I begynnelsen ble ERP system kun brukt av store organisasjoner, siden det var kun de som hadde råd til det. Mindre organisasjoner hadde ikke midlene som trengs for investering innenfor maskinvare, programvare og mennesker for å implementere og støtte ERP system. Etterhvert begynte ERP leverandørene å tilby løsninger med mye lavere oppstartskostnader og raskere og enklere implementasjoner. Sky-baserte løsninger lar kunden unngå dyr ERP programvare og oppgradering av maskinvare, siden de kan leie programvare som kjører på leverandørens maskinvare.

Viktig

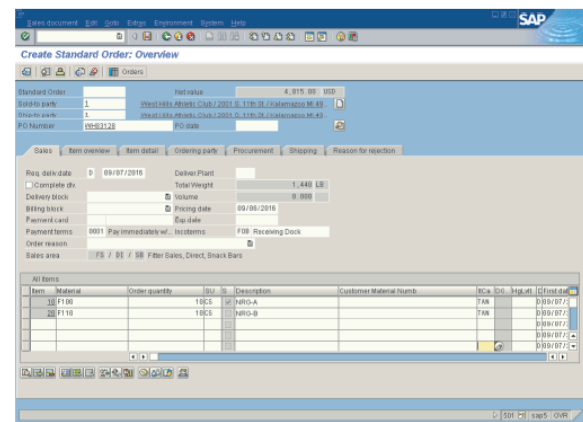
Supply Chain Management (SCM)

Supply Chain Management (SCM) er et enterprise system som inkluderer planlegging, utføring og kontroll av alle aktiviteter involvert i innkjøp og anskaffelse av råmaterialer, omdannelsen av råmaterialer til endelig produkt, lagring i varehus og levering av endelige produkter til kunder. Målet til en SCM er å redusere kostnader, forbedre kundeservicen og redusere inventaret i supply chain. **SCM kontrollerer materialer, informasjon og finans ettersom man beveger seg fra leverandør til produsent til forhandler til selger og tilsatt til kunden.** Bevegelsen av materialer involverer råmaterialer fra leverandør til produsent, samt endelige produkter fra produsent til forhandler, selger og kunde. Bevegelsen av informasjon involverer å lage og sende bestillinger og fakturaer, mens bevegelsen av finans involverer betalingstransaksjoner. Et eksempel på en SCM er systemet som brukes av Posten.



I produksjonsorganisasjoner blir ERP system ofte brukt for å støtte SCM, siden det blant annet øker effektiviteten til interaksjonen med leverandører og partnere, som trengs for å få råmaterialer og ressurser. ERP systemet vil vanligvis omfatte SCM aktiviteter for å kontrollere flyten av materialer, informasjon og finans. Systemet bruker følgende prosess for å utvikle en **produksjonsplan**:

1. **Salgsprognoser** – det lages et estimat for fremtidig etterspørsel fra kunden. Estimaten er ved et høyt nivå og kan gjelde produktgrupper istedenfor enkelte produkttyper. Det kan lages med ERP- eller annen spesialisert programvare.
2. **Salgs- og driftsplan (S&OP)** – bruker etterspørsel, inventarnivå, produksjonskapasitet og sesongvariasjon for å bestemme hvilke produkttyper som må produseres og når de skal tilfredsstillte forventet fremtidig etterspørsel.
3. **Etterspørsel management** – bestemmer ukentlig eller daglig produksjon som trengs for å tilfredsstillte etterspørselen for individuelle produkter. Resultatet er master produksjonsskjema som er en produksjonsplan for alle ferdige goder.
4. **Detaljert planlegging** – lager et detaljert produksjonsskjema som gir hvilke produkter som skal produseres først, hvor lenge produksjonen skal kjøre ved hvert produkt, osv.
5. **Materialbehovsplanlegging (MRP)** – eksisterende inventar og type råmaterialer som trengs for å lage hvert produkt, brukes for å bestemme type og mengde råmaterialer som skal bestilles fra leverandører for å støtte det planlagte produksjonsskjemaet. Mengde råmateriale vil også avhenge av leveringstiden og hvor mye materialer som sendes om gangen fra leverandøren.
6. **Innkjøp** – informasjonen fra MRP brukes til å lage bestillingsordre for råmaterialer fra kvalifiserte leverandører. Bestillingene sendes slik at råmaterialene ankommer når de skal brukes, slik at man minimerer kostnader forbundet med lagring.
7. **Produksjon** – produksjonsskjemaet brukes for å planlegge detaljer ved produksjonsdriften, og det tas hensyn til tilgjengelighet av ansatte, utstyr og råmaterialer, samt detaljert data om kundeetterspørsel.
8. **Salgsbestilling** – aktivitetene som utføres for å lage en kundebestilling, for eksempel registrering av tingene som skal kjøpes, innstilling av salgspriser, registrering av kvantitet, osv. Hvis tingene er utsolgt, må dette kommuniseres til kunden, sammen med anbefalte alternativer. Figuren viser hvordan en bestilling skrives inn i SAP programvare.



Viktig

Customer Relationship Management (CRM)

Customer Relationship Management (CRM) er et enterprise system som hjelper bedriften med å kontrollere alle aspekter ved kundemøter, inkludert markedsføring, salg, distribuering, regnskap og kundeservicen (se figur). CRM system

kan ses på som en adressebok med oversikt over alle interaksjonene som organisasjonen har hatt med hver kunde.

Målet til en CRM er å forstå og forutsi behovene til nåværende og potensielle kunder, for å øke bevaringen og lojaliteten til kunden, samtidig som man optimaliserer hvordan produktene og tjenestene selges. Fordeler ved CRM er at det kan gi bedre kundetilfredshet, øke kundebevaringen, redusere driftskostnader og øke evnen til å tilfredsstillte kundeetterspørselen. **CRM blir mest brukt innenfor salg, markedsføring, distribusjon, regnskap og tjenesteorganisasjoner for å lage og observere data om kunder og forbedre kommunikasjonen.**



CRM programvare automatiserer og integrerer funksjoner innenfor salg, markedsføring og tjenester i en organisasjon. CRM systemet forsøker å forstå handlingene til en kunde ved å lagre data om alle koblingene selskapet har med kunden via de ulike kanalene. Det lages en database om kundene som beskriver forhold med tilstrekkelig detaljer, slik at kundebehov kan matches med produktplaner og tilbud.

Viktige egenskaper ved et CRM system er:

- **Kontakt management** – evnen til å spore data om kunder, og deretter aksessere dataen fra enhver del av organisasjonen
- **Salgsmanagement** – evnen til å organisere data kunder, og deretter prioritere potensielle salgsmuligheter og identifisere neste steg
- **Kundestøtte** – evnen til å støtte de som har ansvar for kundeservice, slik at de kan adressere forespørsler og løse kundeproblemer på en rask, nøye og passende måte samtidig som det blir samlet og lagret data om disse interaksjonene
- **Markedsføring automatisering** – evnen til å analysere alle kundeinteraksjoner, generere passende svar og samle data for å lage effektive markeds kampanjer.
- **Analyse** – evnen til å analysere kundedata for å finne måter å øke salget og redusere kostnader, identifisere «de beste kundene» og bestemme hvordan man skal bevare disse og finne flere.
- **Sosiale nettverk** – evnen til å bli med på sosiale nettverk, slik som Facebook, der man kan komme i kontakt med potensielle kunder og respondere på problemer eller spørsmål kunden har for å forbedre kundetilfredsheten.
- **Aksess fra smarttelefoner** – evnen til å aksessere nettbasert management programvare for kunderelasjoner på mobile enheter
- **Importere kontaktdata** – evnen til å importere kontaktdata fra ulike tjenesteleverandører som kan lastes direkte inn i CRM applikasjonen

CRM handler ikke bare om å installere ny programvare. For å suksessfullt implementere CRM er det essensielt at organisasjonen prioriterer kundene, istedenfor å fokusere på å selge produkter, og dette kan kreve opplæring av ansatte. Det må bestemmes hvem som skal håndtere kunde problemer og når, og relevant informasjon må gjøres tilgjengelig via integrerte datasystem. Tabellen til høyre viser noen kjente CRM system.

Vendor/Product	Select Customers	Pricing Starts At
Infusionsoft CRM	Clean Corp Swim Fitness	\$67 per user/month
OnContact CRM 7	Prudential Carfax	\$50 per user/month
Oracle Marketing Cloud	Bass Pro Shops Whole Foods	\$80 per user/month
Prophet CRM	AT&T Century 21	\$24 per user/month
Sage Software CRM	Panasonic Lockheed Martin	\$39 per user/month
Salesforce Sales Cloud	Dell Dr. Pepper Snapple	\$5 per user/month

Product lifecycle management (PLM)

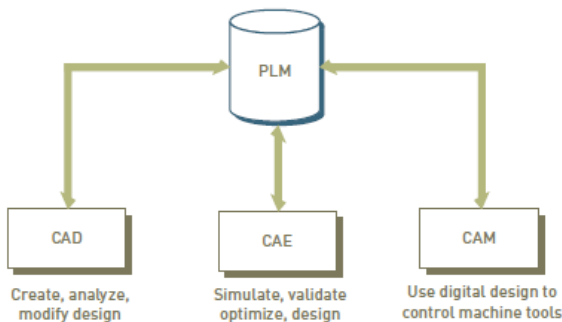
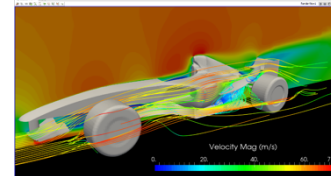
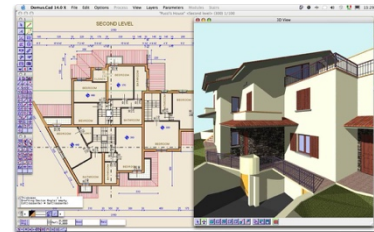
To viktige definisjoner:

- **Product lifecycle management (PLM)** = en enterprise business strategi som lager et felles oppbevaringssted for produktinformasjon og prosesser. Det støtter samarbeidsskapelse, management, formidling og bruk av informasjon om produktet og emballasje.
- **Product lifecycle management (PLM) programvare** = brukes av produksjonsselskap for å kontrollere data og prosesser ved de ulike fasene i livssyklusen til produktet (se figur). Ettersom produkter beveger seg gjennom denne syklusen vil det genereres data som gis til ulike grupper både innenfor og utenfor bedriften.



PLM programvare kan omfatte tre typer programvare:

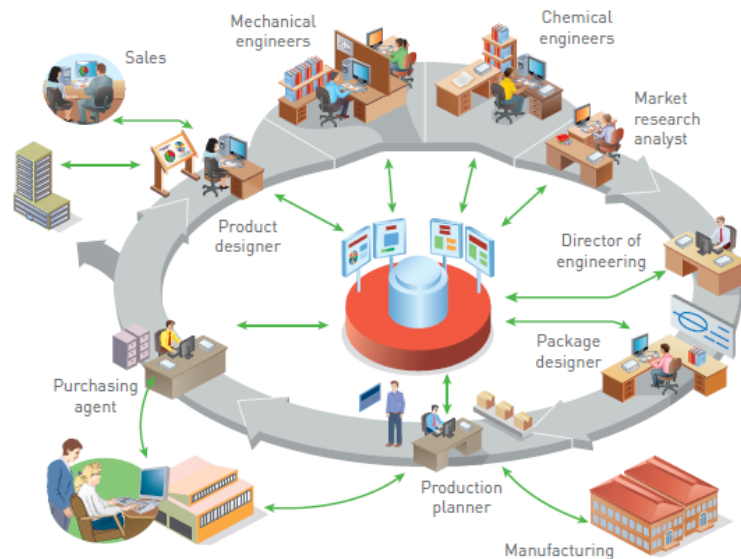
1. **Dataassistert design (CAD)** = bruken av programvare for å hjelpe til i skapelsen, analysen og modifiseringen av designet til en komponent eller produkt. CAD kan brukes for å øke produktiviteten til designeren, øke designkvaliteten og lage en database som beskriver enheten. Dataen kan deles med andre eller brukes i produksjonen.
2. **Dataassistert engineering (CAE)** = bruken av programvare for å analysere robustheten og ytelsen til komponenter og enheter. CAE kan brukes for å simulere, validere og optimalisere produkter og produksjonsverktøy. Dette er svært nyttig for designteam under evaluering og beslutningstaking.
3. **Dataassistert manufacturing (CAM)** = bruken av programvare for å kontrollere maskinverktøy og relatert maskineri under produksjonen av komponenter og produkter.



Modellen som genereres i CAD og verifiseres i CAE kan gis som input til CAM programvare som deretter kontrollerer maskinverktøyene (se figur).

Noen organisasjoner velger å bruke et enkelt, integrert PLM system som omfatter alle de viktigste fasene i livssyklusen til produktet. Andre organisasjoner velger å bruke flere, separate PLM programvarekomponenter fra ulike leverandører, slik at de kan velge programvare som passer best til de ulike fasene i syklusen. Dette tillater inkrementell investering i PLM strategi, men det kan være vanskelig å koble sammen de ulike delene slik at det dannes en enkel, sammenhengende database om dataen og prosessene som skapes i syklusen.

Fordelen ved å bruke PLM strategi er at det fremmer innovasjon og forbedrer produktiviteten ved å koble ansatte til den kunnskapen om produkt og prosess som de trenger for å oppnå suksess (se figur).



Tabellen viser at andre fordeler forbundet med bruken av PLM system er redusert tid før produktet er på markedet, reduserte kostnader og større sikkerhet for at produktene overholder forskrifter.

Benefit	How Achieved
Reduce time to market	<ul style="list-style-type: none"> • By connecting design, research and development, procurement, manufacturing, and customer service seamlessly through a flexible collaboration environment • By improving collaboration among the organization and its suppliers, contract manufacturers, and OEMs
Reduce costs	<ul style="list-style-type: none"> • By reducing prototyping costs through the use of software simulation • By reducing scrap and rework through improved processes • By reducing the number of product components through standardization
Ensure regulatory compliance	<ul style="list-style-type: none"> • By providing a secure repository, tracking and audit trails, change and document management controls, workflow and communications, and improved security

Utfordringer ved implementering av Enterprise system

Implementering av et enterprise system er svært utfordrende, og det krever enorme mengder ressurser, de beste informasjonssystemene og menneskene og mye støtte fra management (eks: opplæring). Mange implementeringer feiler og problemer med enterprise system er dyre å fikse. De største utfordringene ved suksessfull implementering av et enterprise system er:

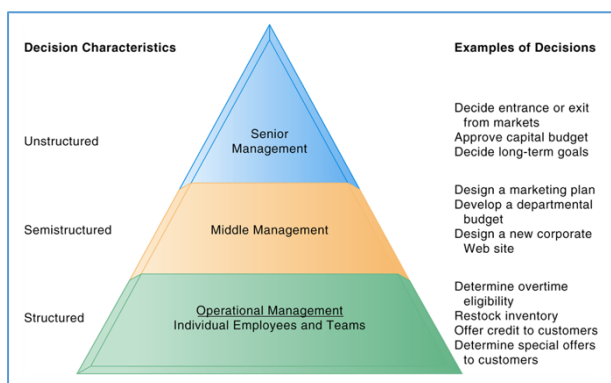
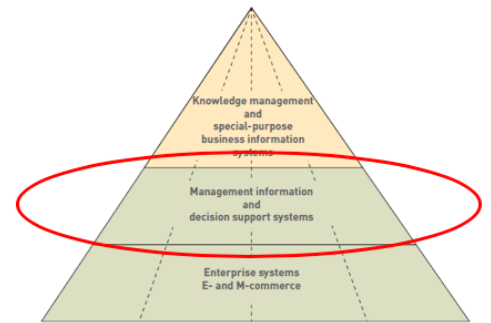
- **Kostnad og forstyrrelser ved oppgraderinger** – de fleste bedriftene har andre system som må integreres med enterprise systemet, slik som e-commerce system, analyseprogram for finans og andre applikasjoner som kommuniserer med leverandører, kunder, osv. Integrering av flere system gjør at implementasjonen tar lengre tid og blir mer kompleks
- **Kostnad og implementeringstid** – gjennomsnittlig implementasjon koster 5,5 millioner dollar og varer over 14 måneder
- **Vanskelige å styre endringer** – bedrifter må ofte utføre radikale endringer i hvordan de opererer for å tilpasse seg de nye arbeidsprosessene. Disse endringene kan være så drastiske at erfarne ansatte velger å forlate bedriften istedenfor å tilpasse seg.
- **Kontroll over programvaretilpasning** – enterprise systemet må kanskje tilpasses slik at det tilfredsstiller de essensielle kravene til bedriften, noe som kan bli svært dyrt og kan forsinke implementasjonen
- **Brukerfrustrasjon over nytt system** – for å effektivt bruke enterprise systemet må arbeidsprosesser endres, og mange brukere kan i begynnelsen bli frustrerte over disse endringene og behøver kostbar opplæring og motivering.

Følgende er tips for å overkomme disse utfordringene:

1. Finn en heltidsansvarlig som styrer prosjektet
2. Bruk en uavhengig og erfaren ressurs som gir prosjektoversikt og verifiserer ytelsen til systemet
3. Gi tilstrekkelig tid til overgangen fra gammelt til nytt system
4. Bruk tilstrekkelig med tid og penger på opplæring av ansatte
5. Definer måter å vurdere prosjektprogresjon og identifiser prosjektrelaterte risikoer
6. Sørg for at omfanget til prosjektet er godt definert
7. Vær forsiktig med å endre enterprise systemet slik at det skal passe til bedriftens praksiser (heller omvendt!)

Kapittel 9 – Business Intelligence og Analytics

Vi lever i en tid med big data, der ny data flyter over oss fra alle retninger ved en enorm hastighet. I den siste tiden har vi fått verktøy og forståelse for hvordan vi kan bruke denne dataen til noe nyttig. Organisasjoner har lært seg å analysere stor mengde data, ikke bare for å måle tidligere og nåværende ytelse, men også for å forutsi fremtiden. Bedrifter trenger enklere og raskere måter å oppdage relevante mønstre og innsikt i dataen for å bedre støtte beslutningstaking. Bedrifter kan ha tilgang til samme data, men bedrifter som raskere analyserer dataen kan få konkurransefordeler. **Business Intelligence og Analytics brukes innenfor nesten alle fagområder (eks: finans, helse, markedsføring, osv.) for å sikre bedre beslutningstaking.**



Figuren viser de tre nivåene for management i en organisasjon: senior, middel og operasjonell management. Disse har ulike egenskaper og er ansvarlig for ulike typer beslutninger. Middel management er ansvarlig for å følge med på hvordan virksomheten fungerer. Dette innebærer informert problemløsning, der man må handle når virksomheten ikke fungerer som den skal (eks: for lite produksjonsvolum, tidsfrister blir ikke nådd, for høy kostnadsøkning, osv.).

Introduksjon – Business Intelligence og Analytics

Business analytics er ekstensiv bruk av data og kvantitativ analyse for å støtte faktabasert beslutningstaking i en organisasjon. Det kan brukes for å få bedre forståelse over forretningsytelse, finne nye forretningsmønstre og relasjoner, forklare hvorfor bestemte resultat oppstår, optimalisere nåværende drift og forutsi fremtidige resultater.

Business Intelligence (BI) omfatter en rekke applikasjoner, praksiser og teknologier for uthenting, transformering, integrering, visualisering, analyse, tolkning og presentasjon av data for å støtte beslutningstaking. Dataen som brukes i BI blir ofte hentet fra flere kilder, som kan være interne eller eksterne til organisasjonen. **Denne dataen kan brukes for å lage store samlinger av data, som kalles datavarehus, databutikker eller datasjøer.** Brukere kan være ansatte, kunder, leverandører og partnere, og de får tilgang til dataen og BI applikasjonene via nettet (intra- eller ektranett) og mobile enheter. **Målet til BI er å få mest mulig verdi ut av informasjon og presentere resultatet av analysen på en enkel og forståelig måte.**

Fordeler ved BI og Analytics

BI og Analytics kan brukes for å:

- **Detektere svindel** = samle data om mistenksomme anmerkninger, slik at det blir lettere å oppdage bedrageri
- **Forbedre prognoser** = forutsi etterspørsel, slik at inventaret kan justeres deretter
- **Øke salg** = optimalisere avgjørelser innenfor prissetting, noe som kan øke salget.
- **Optimalisere driften** = bestemme hvilke produkter som skal produseres basert på analyser av markedet
- **Redusere kostnader** = optimalisere system, slik at kostnader kuttes

Komponenter i BI og Analytics

For at organisasjonen skal få verdi ut av BI og Analytics, må følgende komponenter være på plass:

- **Et solid datahåndteringsprogram med datastyring.** Datahåndtering er et sett med funksjoner som definerer prosesser der data blir samlet inn, lagret, strukturert og behandlet for å sikre at dataen er tilgjengelig, pålitelig og tidsriktig. Datastyring er kjernen av datahåndtering, og det definerer roller, ansvar og prosesser som trengs for å sikre at dataen er pålitelig og kan brukes av hele organisasjonen. Det blir identifisert personer som er ansvarlig for å fikse og unngå problemer med dataen.
- **Kreative datavitenskapsmenn**, som forstår forretningen og begrensninger ved deres data, verktøy og teknikker. Dette brukes for å forbedre beslutningstakingen.
- **Management team med sterk forpliktelse til datadreven beslutningstaking.** De nødvendige komponentene blir hurtig satt sammen for å utføre gode beslutninger som sikrer konkurransefordel.
- **Teknologisk infrastruktur.** Organisasjonen må ha tilgang til riktig programvare, maskinvare og nettverk



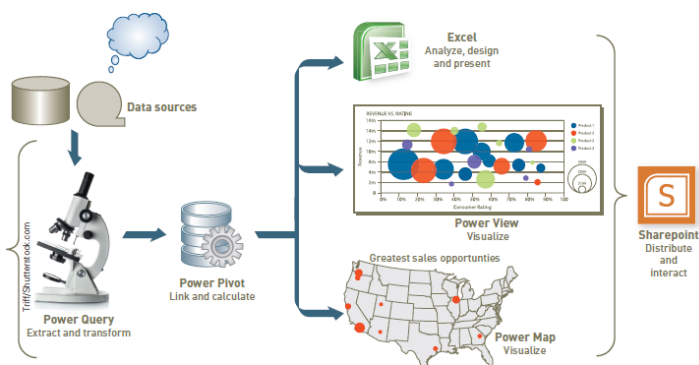
Verktøy innenfor Business Intelligence og Analytics

Vi skal se nærmere på noen av verktøyene som brukes innenfor Business Intelligence og Analytics.

Spreadsheets (regneark)

Bedriftsledere vil ofte importere data inn i regnearkprogram, slik som Excel, som deretter utfører operasjoner på dataen basert på formler brukeren har gitt. Det kan også lages grafer eller diagrammer basert på dataen. Microsoft Power BI bruker Excel-programvare for regneark, og det har blant annet blitt brukt for å bedre forstå klinisk bruk av medisiner,

effektivitet til behandling og assosierte kostnader (se figur). Power BI modellen blir gitt interessant data, for eksempel om helse, vær, lokasjoner, behandling, osv., og den produserer grafer, visualiserer data om regionale kart, osv. Modellen finner koblinger mellom forskjellige datasett og gir dermed innsikt som ellers kunne vært vanskelig å oppnå. For eksempel kunne man se hvordan været påvirket frekvensen av lungesykdommer.



Rapportering- og spørringsverktøy

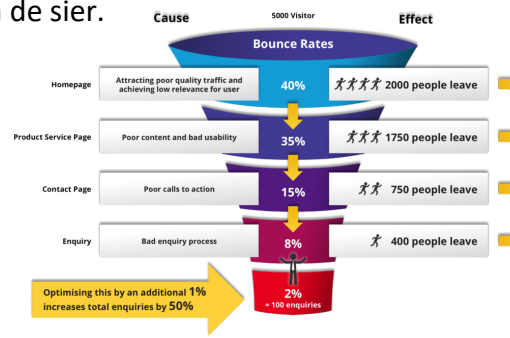
Mange organisasjoner har investert i rapporteringsverktøy som lar ansatte få data de trenger for å løse problemer eller identifisere muligheter. **Rapportering- og spørringsverktøy kan presentere denne dataen på en måte som er lett å forstå vha formatert data, grafer og diagrammer.** Disse verktøyene lar brukeren lage egne spørringer og formatere resultatet uten hjelp fra IT organisasjonen. For eksempel blir QlikView brukt av FFF Enterprises for å spore og kontrollere volum og verdien til alle produkttransaksjoner.

Data visualiseringsverktøy

Ved data visualisering blir data presentert i et billedlig eller grafisk format. De fleste menneskene er flinkere til å se signifikante trender, mønster eller relasjoner i dataen dersom den presenteres i et grafisk format istedenfor tabeller eller regneark. Beslutningstakere er positive til programvare som tilbyr data visualisering. **Representasjon av data i visuell form**

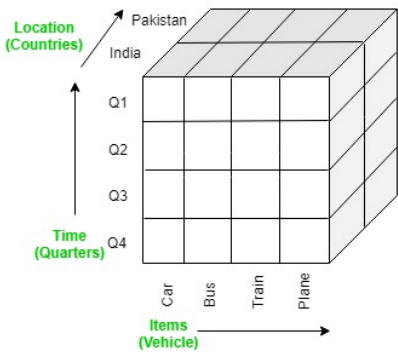
gir øyeblikkelig innvirkning på kjedelige tall. Det finnes mange verktøy for å skape visuelle representasjoner som kan avsløre mønstre eller forhold i dataen som ellers er vanskelig å oppfatte, for eksempel:

- **Ordsdy** = en visuell skildring av et sett med ord som har blitt gruppert sammen basert på hvor ofte de forekommer. Kan brukes for finne ut hvem som snakker om et produkt og hva de sier. Tekst fra dokumenter eller nettsider blir brukt for å telle antall ganger et ord eller uttrykk blir nevnt.
- **Konversjonstunnel** = en grafisk representasjon som summerer hvilke steg forbrukeren tar når de avgjør om de skal kjøpe et produkt og dermed bli kunde. De kan brukes for å identifisere steg som gjør at forbrukere blir forvirret.



Online analytisk prosessering (OLAP)

Online analytisk prosessering (OLAP) brukes for å analysere multidimensjonal data fra mange ulike perspektiver. OLAP kan brukes for å utføre trendanalyser og identifisere problemer og muligheter. Databaser som støtter OLAP består av **datakuber**, som inneholder numerisk fakta som kalles målinger og er kategorisert etter dimensjon (eks: tid og geografi). Et eksempel er en datakube, der den ene målingen er salget av et bestemt produkt. Denne verdien blir vist langs *Metric* dimensjonen, mens tidsdimensjonen kan være en spesifikk dag og geografidimensjonen kan være en spesifikk butikk.



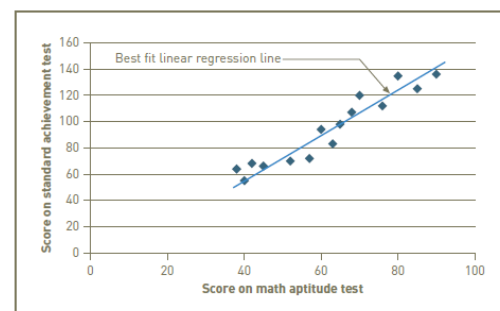
OLAP kan brukes for å komprimere data i summeringer som gjør det lettere å analysere dataen for å svare på ulike spørsmål. For eksempel kan det summere salget av en bestemt enhet på en spesifikk dag i en spesifikk butikk, eller summere salget av en bestemt enhet på en spesifikk dag i alle butikkene. OLAP kan også lage datakuber med flere dimensjoner. De kan brukes for å hjelpe organisasjoner med å forutsi etterspørsel og maksimere salg (eks: Starbucks).

Drill-down analyse

Drill-down analyse gir innsikt i detaljene ved dataen som beslutningstakere trenger for å forstå hvorfor noe har skjedd, siden store problemer ofte kan skyldes at små ting ikke har gått som planlagt. **Ved drill-down analyse blir høy-nivå data utforsket med økende detaljgrad for å få innsikt i bestemte elementer.** En analogi er å trinnvis fjerne hvert lag av en løk til man når kjernen. Et eksempel er når en salgsdirektør ønsker å vurdere det verdensomspennende salget i løpet av siste kvartal. Først ser man på salget i hvert land, videre drilling utføres for å se på salget i et bestemt land ila siste kvartal, videre drilling utføres for å se salget i et bestemt land ila en bestemt måned og videre drilling utføres for å se salget til en bestemt produktlinje i et bestemt land ila en bestemte måned.

Lineær regresjon

Lineær regresjon er en matematisk teknikk der verdien til en avhengig variabel kan forutses basert på en enkelt uavhengig variabel og det lineære forholdet mellom de to. Gitt et sett med verdier for den avhengige og uavhengige variabelen, kan lineær regresjon brukes for å finne den rette linjen som passer best med observasjonene. Minste-kvadratets metode er den mest brukte metoden for å finne regresjonslinjen. **Lineær regresjon sier ikke at en verdi forårsaker den andre, men heller at når en verdi øker vil den andre verdien også øke eller minke proporsjonalt mye.**



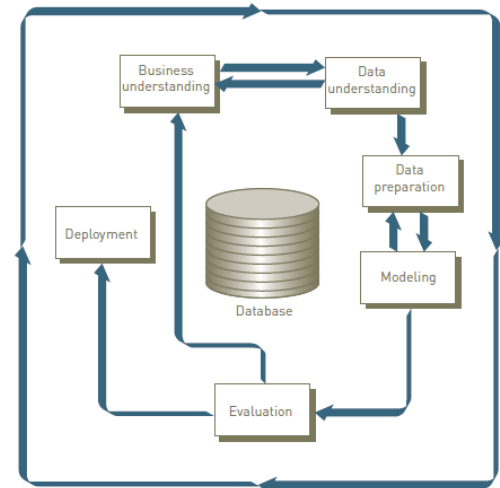
Data Mining

Data Mining er et BI-analyseverktøy som brukes for å oppdage skjulte mønster i store datamengder, slik at man kan forutsi fremtidige trender og atferd, noe som er nyttig i beslutningstaking. Dette kan brukes for å forutsi hva som vil skje, slik at managere kan være forberedte til å utnytte muligheter eller unngå problemer. De tre vanligste teknikkene innenfor Data Mining er:

- **Assosiasjonsanalyse** = et spesialisert sett med algoritmer sorterer data og lager statistiske regler om tingenes relasjoner
- **Neural databehandling** = mønster i historisk data brukes for å lage prediksjoner
- **Case-basert resonnering** = historiske if-then-else tilfeller brukes for å gjenkjenne mønster

Cross-Industry Process for Data Mining (CRISP-DM) er en seks-faset, strukturert tilnærming for planlegging og utføring av et Data Mining prosjekt. CRISP-DM er den mest brukte metoden for Data Mining prosjekter, og den er godt bevist og robust. De seks fasene i CRISP-DM er:

1. **Business forståelse** = klargjøring av forretningsmålene til data mining prosjektet og design av prosjektplan som vil oppnå disse målene
2. **Data forståelse** = samle dataen som skal brukes, bli kjent med dataen og identifiser eventuelle problemer ved datakvaliteten og avgjør hvordan de skal håndteres.
3. **Data preparasjon** = velg et subsett av data som skal brukes, håndter kvalitetsproblemer og transformer dataen til en form som passer analyse.
4. **Modellering** = bruk valgte modelleringsteknikker
5. **Evaluering** = vurder om modellen oppnår forretningsmålene
6. **Utplassering** = bruk modellen i organisasjonens beslutningstaking



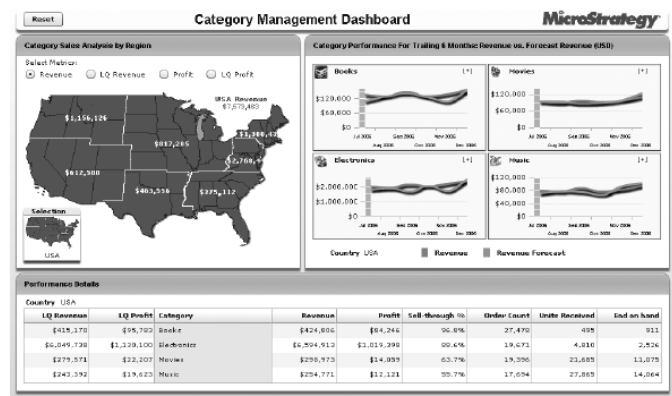
Noen eksempler som viser hvordan data mining kan brukes er:

- Identifisering av forbrukere som sannsynligvis vil dra nytte av fremtidige salgsfremmende mailer, ved å se på tidligere responser til slike mailer.
- Undersøke handelsdetaljer for å identifisere produkter som er tilsynelatende urelaterte, men som ofte kjøpes sammen
- Bruke hotellbestillingsdata for å justere romprisene slik at fortjenesten blir maksimert
- Identifisering av kunder som er mest lønnsomme å rekruttere ved å analysere demografisk data og atferdsdata om potensielle kunder
- Gjenkjenne hvordan endringer i DNA sekvenser til individer kan påvirke risikoen for å utvikle vanlige sykdommer som Alzheimer eller kreft

Dashboards

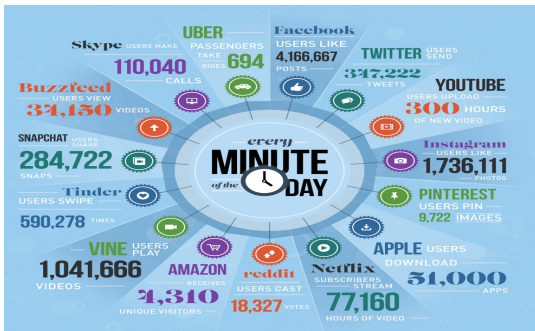
Under utføringen av en strategi for å oppnå organisasjonens mål, kan en KPI (Key Performance Indicators) brukes for å observere fremdriften. For å kunne sammenligne over ulike tidsperioder, er det viktig at KPIs blir definert og at samme definisjon brukes over periodene. Noen KPIs kan bli droppet og andre kan legges til ettersom organisasjonen får nye mål. Et eksempel på en KPI er innenfor kundeservice, der antall kundesamtaler som

besvares innen de første fire ringingene skal økes med 90 % ilt de neste tre månedene. **Et dashboard kan brukes for å vise frem et sett med KPIs som beskriver tilstanden til en prosess ved et bestemt tidspunkt. De brukes for å gi hurtig tilgang til informasjon på en presis og lett forståelig måte, slik at organisasjonen kan kjøre mer effektivt. Brukere ved hvert nivå av organisasjonen får informasjon som de trenger for å gjøre bedre beslutninger.** KPIs kan vises som kart, trendlinjer, spredningsdiagram, målere, osv. Operasjonelle dashboards kan hente data i sanntid fra ulike kilder, slik som bedriftsdatabaser og spreadsheets, slik at beslutningstakere får tilgang til tidsriktig data.



Big data

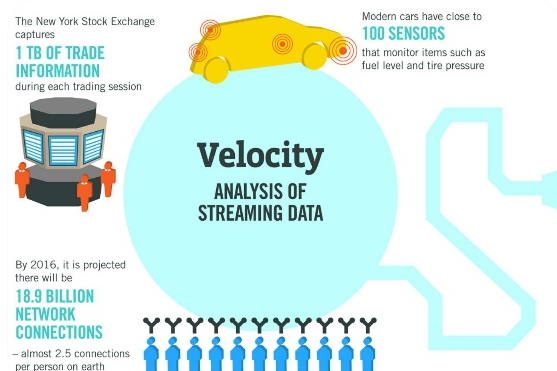
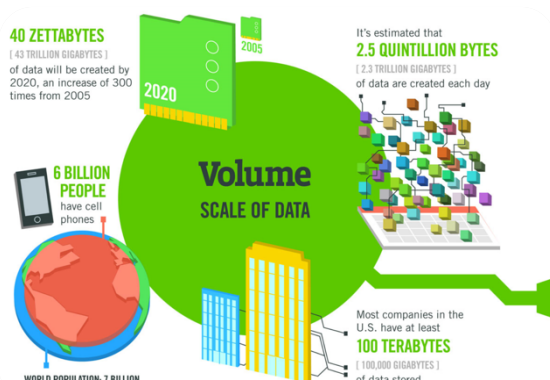
Big data analytics er et fagområde der store og varierte datasett blir undersøkt for å finne ukjent korrelasjoner, markedstrender, kundepreferanser og annen informasjon som kan hjelpe organisasjonen til å ta bedre beslutninger. Det er altså bruken av data med høy volum, mye variasjon og høy hastighet for å generere handlingsdyktig innsikt. Big Data Analytics blir kombinert med flere andre fagområder, slik som Deep Learning, Nevrale nettverk, maskinlæring, AI, osv. Big data vil lede den neste digitale transformasjonen. **79% av alle bedriftsledere sier konkurransevne og bli utsatt for utryddelse.**

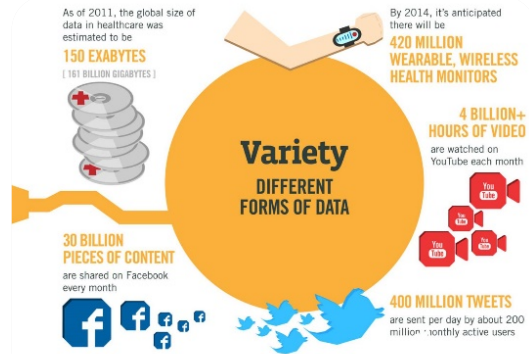
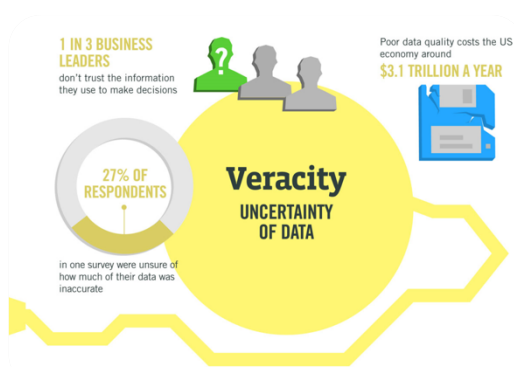


Big data brukes for å beskrive datasett som er så store og komplekse at de ikke kan behandles av tradisjonelle programvarer for dataprosessering. CEO for Google har sagt at i løpet av to dager vil vi skape like mye informasjon som vi gjorde i perioden fra sivilisasjonens start til 2003. 1.5 billioner mennesker besøker Facebook hver dag og bruker i gjennomsnitt 135 minutter per dag. Innen 2020 vil hvert menneske i gjennomsnitt produsere 1.7 MB data per sekund.

Det er fire spesifikke attributter, kalt de fire V'ene, som bestemmer om data kan defineres som Big Data:

- **Volume** = hovedegenskapen som gjør at dataen er «big» er volumet. Big data har et stort volum som enten bruker mye lagringsplass eller består av et stort antall register
- **Velocity** = reflekterer hvilken hastighet dataen blir samlet inn og analysert. For Big Data er det en høy frekvens av innkommende data som må prosesseres
- **Veracity (pålitelighet)** = refererer til hvor pålitelig dataen er. Kvaliteten til Big Data kan variere, noe som påvirker den nøyaktige analysen
- **Variety** = variasjonen til Big Data er stor, siden dataen hentes fra flere strukturerte og ustrukturerte kilder

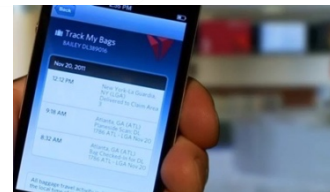




Big data – verdiskaping og eksempler på bruk

Big Data kan skape verdi på følgende fem måter:

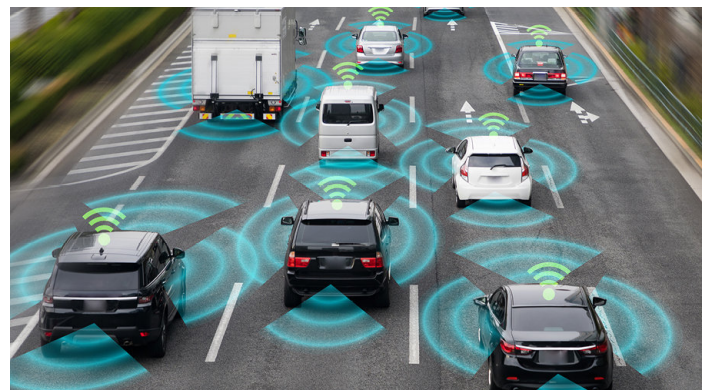
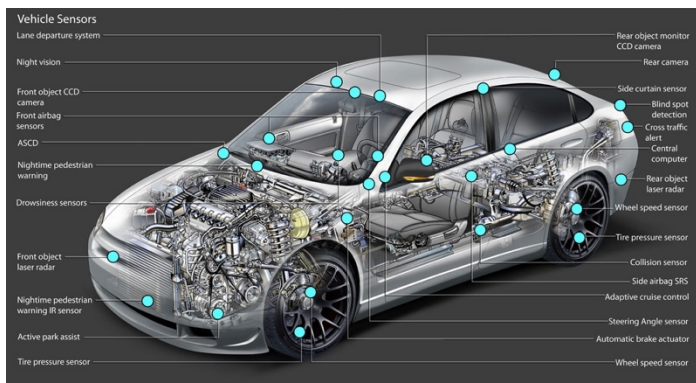
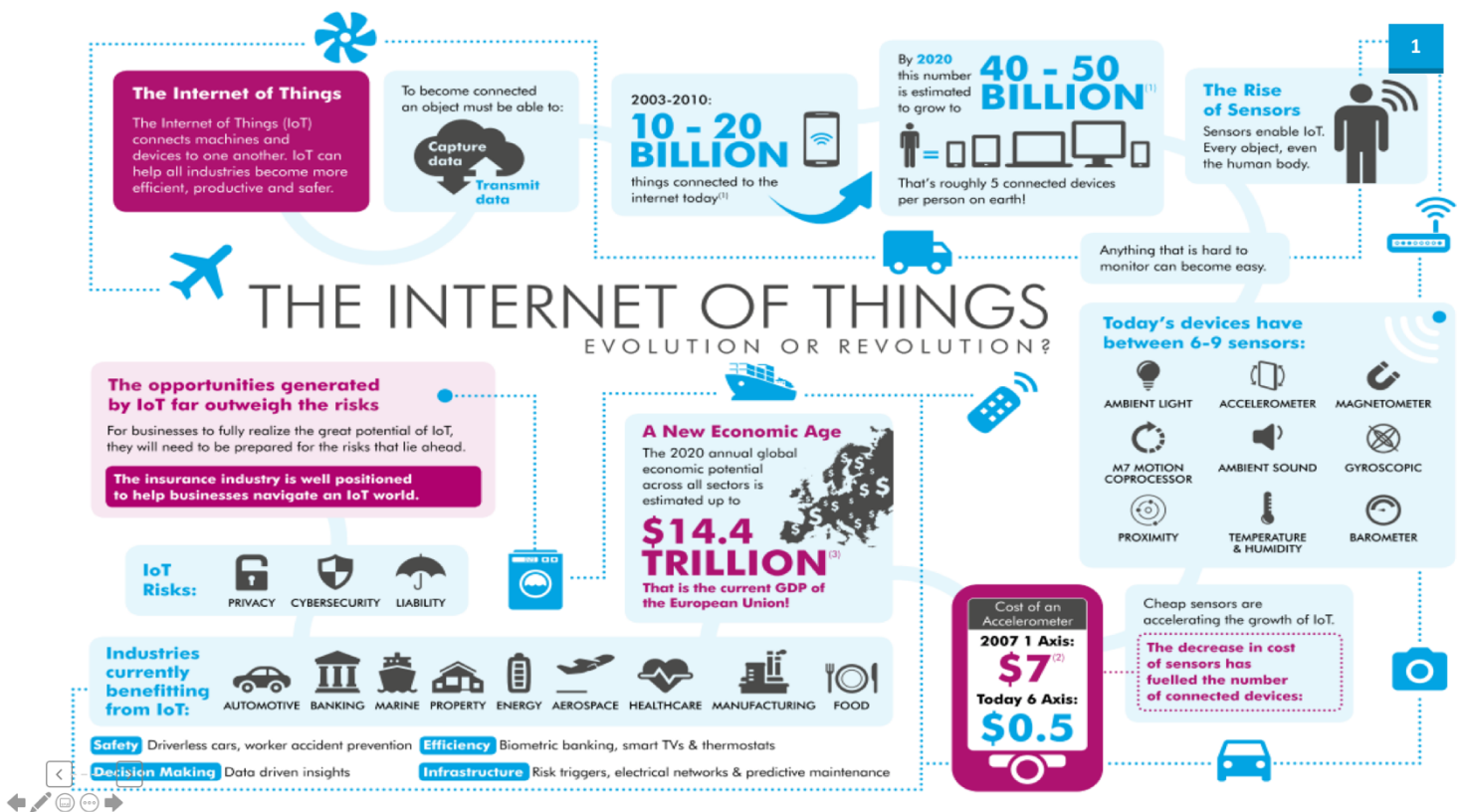
- Transparent (åpenhet)** = Big Data gjør at tidsriktig informasjon blir lettere tilgjengelig for relevante interessenter, ved at det fungerer som en ny form for presentering av informasjon. Et eksempel er Delta Airlines, som var en av de første flyselskapene som lot kundene spore bagasjen sin i sanntid fra mobile enheter, ved å legge til RFIDs i bagmerker. Siden de behandler over 130 millioner bagger per år vil dette representere en stor mengde data. Appen har blitt lastet ned over 11 millioner ganger og den bidrar til å forbedre kundetilfredsheten.
- Beslutningstaking** = Big Data kan brukes for å forbedre beslutningstaking, minimere risikoer og avdekke verdifull innsikt som ellers ville forblitt skjult. Det legger opp til bedre informerte beslutninger i sanntid. Et eksempel er transportfirmaet UPS, som bruker spesialiserte programmer for å finne optimaliserte ruter, forutsi vedlikehold, osv. UPS leverer mer enn 18 millioner pakker og dokumenter per dag, og over 4 billioner enheter per år vha nesten 100 000 kjøretøy. Etter å ha tatt i bruk programmet har selskapet spart over 148 millioner liter drivstoff, unngått å kjøre 585 millioner kilometer og redusert tomgangstid med 206 millioner minutter.
- Markedsføring** = Big Data lar organisasjoner bli bedre kjent med kunden, slik at de kan lage svært spesifikk segmentering og deretter skreddersy produkter og tjenester til segmentene slik at de tilfredsstiller deres behov. Et eksempel er Amazon som utfører en segmentering av populasjonen for å skreddersy tilbud og markedsføring. Dette førte til en økning i salget på 29%. Algoritmen er basert på kundens kjøperhistorikk, ting i handlekurven, ting de har vurdert og likt og hva andre kunder har sett på og kjøpt.
- Ytelse** = Big Data lar organisasjoner samle mer nøyaktig og detaljert data om ytelse i sanntid, slik at ytelsen kan finjusteres. Et eksempel er Southwest Airlines som bruker taleanalyse for å hente ut data fra liveopptak av interaksjoner mellom kunder og personell, for å bedre forstå deres kunder. Spørsmål og viktig data fra kunden blir vist frem slik at interaksjonen kan finjusteres for å gi bedre ytelse og effektivitet.
- Innovasjon** = Big Data lar bedrifter lage nye produkter og tjenester, forbedre eksisterende og oppdage nye forretningsmodeller. Et eksempel er BlackRock Investment Group som bruker en rekke teknologier, inkludert Big Data og maskinlæring, for å systematisk velge aksjer og se



«investeringssignaler» fra en rekke informasjonskilder (eks: internettrafikk og satellittbilder). De analyserer trafikken på nettsiden til et selskap og bruker dette som en indikator på den fremtidige veksten til selskapet.

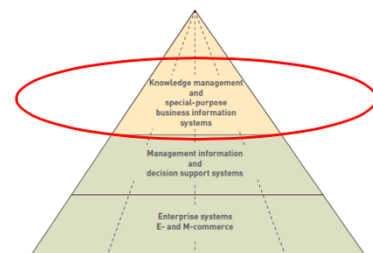
Big Data og IoT

Internet of Things (IoT) er nettverket av enheter, kjøretøy og hvitevarer som inneholder elektronikk, programvare, aktuatorer og konnektivitet som gjør at disse kan kobles til, interagere og utveksle data. IoT handler om enheter, data og konnektivitet, og den virkelige verdien ved IoT er at det skaper smartere produkter som leverer intelligent innsikt og nye funksjoner. **Ettersom millioner av enheter kobles sammen, vil IoT utløse en enorm innstrømming av Big Data.** Disse fagfeltene kan kombineres for å hente data fra sensorer og deretter bruke dataen for å oppdage mønster og korrelasjoner i sanntid. Eksisterende teknologier innenfor Big Data må forbedres slik at de kan effektivt lagre, styre og hente verdi fra kontinuerlig strøm av sensordata fra enheter i IoT. **For eksempel er det estimert at tilkoblede biler vil sende 25GB med data til skyen per time fra 200 sensorer.** Den største utfordringen er å finne verdien ved denne dataen og bruke den for å utlede handlingsbaserte hendelser.



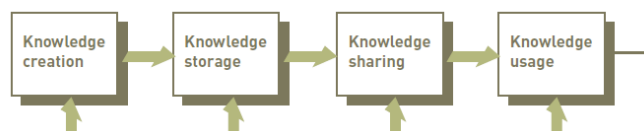
Kapittel 10 – Knowledge Management og spesialiserte IS

Knowledge Management og spesialiserte informasjonssystem blir brukt i nesten alle industrier. Knowledge Management system kan brukes for å få råd om hvordan man skal tilnærme seg et problem som andre i organisasjonen allerede har løst. Spesialiserte informasjonssystem inkluderer nevrale nettverk som brukes for å oppdage mønstre under investering, datasimulering som brukes i opplæring av militæret, ekspertsystemer som bestemmer hvor man skal søke etter olje og gass, osv.

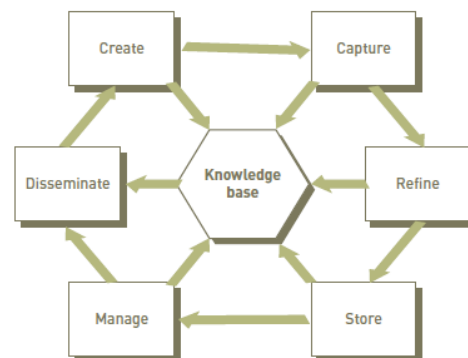


Knowledge Management

Knowledge Management (KM) omfatter en rekke praksiser som er opptatt av å øke bevisstheten, fremme læring, fremskynde samarbeid og innovasjon og utvikle innsikt. Det brukes av organisasjoner for å la individer, teams og hele organisasjonen kollektivt og systematisk lage, lagre dele og bruke kunnskap for å oppnå mål (se figur over).



Som vi kan se på figuren til venstre er **KM system en organisert samling av mennesker, prosedyrer, programvare, databaser og enheter som brukes for å skape, fange, raffinere, lagre, styre og spre kunnskap og erfaring.**



Husk forskjellen mellom:

- **Data** = rå fakta
- **Informasjon** = samling av fakta som er organisert slik at de får større verdi
- **Kunnskap** = bevissthet og forståelse av et sett med informasjon og hvordan informasjonen kan brukes for å støtte en spesifikk oppgave eller ta en beslutning

Kunnskap kan klassifiseres som:

- **Eksplisitt kunnskap** = objektiv kunnskap som kan måles og dokumenteres i rapporter, papirer og regler. Eksempler er standard prosedyrer og produktformler
- **Taus kunnskap** = kunnskap som enkelte har utviklet pga. erfaring. Den er vanskelig å måle og dokumentere, men den er essensiell for høy ytelse og konkurransefordel siden det er vanskelig for andre å kopiere. Den er som regel ikke objektiv eller formel, og den kan involvere faktorer slik som tro, perspektiv og verdisystem. Eks: sykling, diagnostisering av sjeldne sykdommer, osv.

Mye av den tause kunnskapen til ansatte er ekstremt nyttig, men vanskelig å dele med andre. **Et viktig mål ved Knowledge Management er å fange og dokumentere denne verdifulle tause kunnskapen, og omforme den til eksplisitt kunnskap som kan deles med andre.** Dette er enklere sagt enn gjort! Ekspertene kan utvikle egne prosesser, og noen ganger kan disse bli så internaliserte at de ikke klarer å dokumentere prosessene som steg-for-steg instruksjoner. **To metoder som ofte brukes for å fange taus kunnskap er:**

1. **Shadowing** = en nybegynner vil observere hvordan ekspertene utfører jobben sin for å lære prosessen.
2. **Felles problemløsning** = nybegynneren og ekspertene arbeider sammen for å løse et problem, slik at ekspertens fremgangsmåte blir sakte, men sikkert avslørt til den observante nybegynneren

Asset Type	Description	Examples
Explicit knowledge	Knowledge that is documented, stored, and codified	Customer lists, product data, price lists, a database for telemarketing and direct mail, patents, best practices, standard procedures, and market research results
Tacit knowledge	Personal knowledge is not documented but embedded in individual experience	Expertise and skills unique to individual employees, such as how to close a sale or troubleshoot a complex piece of equipment

Bruksområder for Knowledge Management (Ikke i F)

Noen bruksområder for Knowledge Management (KM) er:

- **Fremme innovasjon** = KM kan brukes for å fremme fri flyt av ideer blant ansatte, entreprenører, leverandører og andre business partnere. Dette kan føre til oppdagelser av nye muligheter, som etter evaluering og testing kan føre til økt fortjeneste, reduserte kostnader eller skapelsen av nye produkter og tjenester.
- **Utnytte ekspertise på tvers av organisasjonen** = KM kan legge til rette for at ansatte kan dele og bygge på hverandres erfaring og ekspertise. Nye ansatte eller ansatte som flyttes til nye posisjoner, kan raskere finne seg til rette. Man hindrer at ansatte gjentar feil som andre har gjort tidligere. Dette gjør at resultatet kan leveres raskere, produktiviteten øker og produkter og ideer kan entre markedet tidligere.
- **Fange kunnskapen til viktige individer før de slutter** = KM kan hindre at organisasjoner mister verdifull erfaring og ekspertise som følge av at ansatte slutter

Implementering av Knowledge Management (Ikke i F)

Noen viktige steg for å implementere Knowledge Management er:

1. **Koble KM til organisasjonens mål** – det er viktig å klargjøre hvordan KM systemet vil støtte oppnåelsen av spesifikke mål hos organisasjonen, for eksempel øke fortjeneste, redusere kostnader, forbedre kundeservice, osv. Dette vil gjøre det enklere å selge prosjektet til andre og øke deres støtte og entusiasme.
2. **Start med et pilotprosjekt som involverer entusiaster** – et prosjekt i liten skala er enklere å kontrollere, og dersom utfallet ikke er suksessfullt, vil det ikke påvirke organisasjonen i stor grad. Det er også lettere å få ressurser for små prosjekter, og sannsynligheten for suksess er større når prosjektet involverer entusiaster.
3. **Identifiser verdifull taus kunnskap** – all taus kunnskap er ikke like verdifull, så derfor må man bestemme hvilken kunnskap som skal prioriteres.
4. **Engasjer ansatte** – managere må lage en arbeidskultur som verdsetter taus kunnskap og oppfordrer ansatte til å dele den.

Teknologier som støtter Knowledge Management

Det er en rask økning i mengde kunnskap som er tilgjengelig, og denne kunnskapen må filtreres og distribueres slik at brukerne får tilgang til relevant og tidsriktig kunnskap med høy kvalitet. Derfor trengs det teknologi for å produsere, lagre, distribuere, integrere og kontrollere denne kunnskapen. Vi skal se nærmere på teknologi som støtter KM.

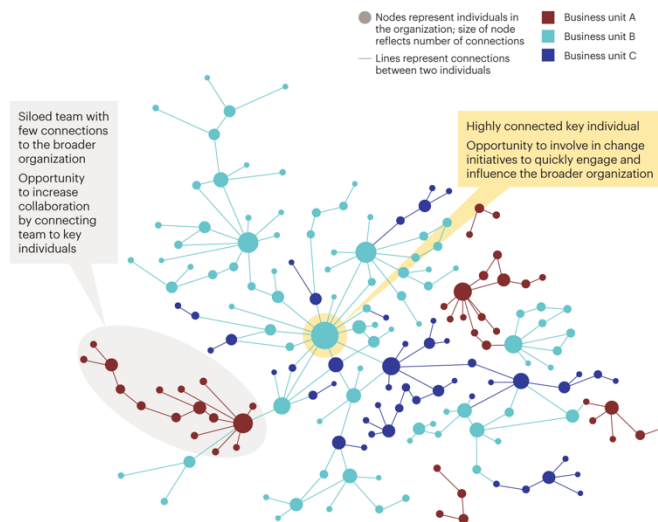


A screenshot of the Sermo website showing a community of practice interface. The page has a navigation bar with 'Home', 'Postings', 'Physicians', 'Journals and News', and 'My Account'. Below the navigation bar, there is a 'Welcome to Sermo' message and a 'Featured Postings' section with a table of posts. The table has columns for Title, Status, Category, Age, and Rating. There are also 'sermo inside' and 'sermo buzz' sections on the right side of the page.

Communities of Practice (CoP)

En CoP (Community of Practice) er en gruppe der medlemmene deler mål og interesser, og de samles for å lage, lagre og dele kunnskap om et spesifikt emne. Et praksisfellesskap blir utviklet rundt emner som er viktige for medlemmene. En CoP vil som regel utvikle ressurser slik som modeller, verktøy, dokumenter, prosesser og terminologi som representerer den akkumulerte kunnskapen til fellesskapet. Det er ikke unormalt at en CoP kan ha medlemmer fra ulike organisasjoner. CoP regnes som en teknologi som støtter KM, siden det er en måte å utvikle ny kunnskap, fremme innovasjon eller dele eksisterende taus kunnskap innenfor organisasjonen.

Illustrative example of a company's informal network

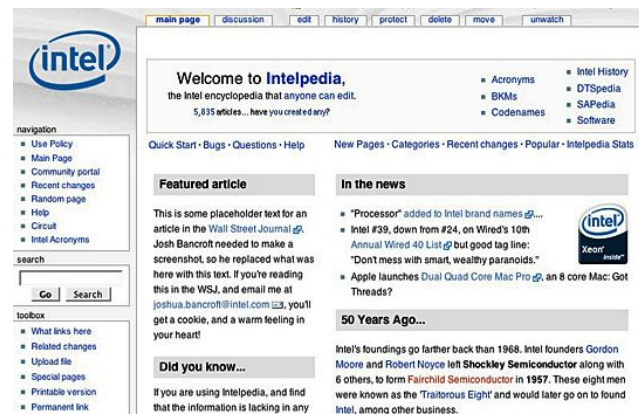


Organisasjon nettverksanalyse (ONA)

Organisasjon nettverksanalyse (ONA) er en teknikk som brukes for å dokumentere og måle flyt av informasjon blant individer, arbeidsgrupper, organisasjoner, datamaskiner, nettsider og andre informasjonskilder (se figur). Hver node i diagrammet er en kunnskapskilde, mens hver kobling representerer flyt av informasjon mellom to noder. **ONA har mange bruksområder, for eksempel kartlegging av kunnskapsflyten, identifisering av kunnskapsmangel i organisasjonen, etablering av samarbeidsnettverk og identifisering av eksperter som kan dele kunnskap.**

Web 2.0 teknologier

Web 2.0 er et begrep som beskriver endringer i teknologi og nettsidedesign som utføres for å fremme informasjonsdeling, samarbeid og funksjonalitet på nettet. Eksempler på Web 2.0 teknologier er blogg, forum, podcast og Wiki. Disse støtter Knowledge Management og bidrar dermed til bedre samarbeid, deling av kunnskap og bygging av bedriftsminne.



Business rule management system (BRMS) (ikke i F)

Business rule management system (BRMS) er programvare som brukes for å definere, utføre, overvåke og opprettholde beslutningslogikken som brukes av de operasjonelle systemene og prosessene som kjører i organisasjonen. Beslutningslogikken involverer retningslinjer, krav og påstander som bestemmer hvordan systemer virker, og når virksomheten er i endring må disse også oppdateres. Det er viktig at organisasjonen er raske til å reagere på endringer, slik at de kan forbli konkurransedyktige. BRMS brukes for å unngå flaskehalser og forsinkelser i implementering av endringer og gjør endringene mer nøyaktige.

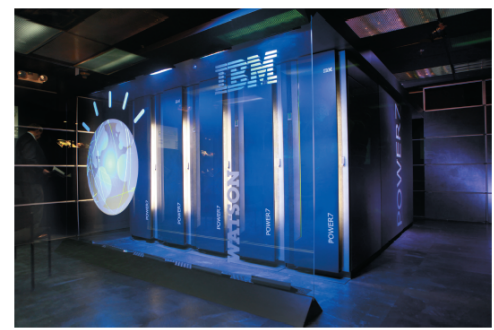
Enterprise søkeverktøy (ikke i F)

Enterprise søk er bruken av søketeknologi for å finne informasjonen innenfor en organisasjon. Enterprise søkeprogramvare vil matche spørsmålet til brukeren opp mot mange informasjonskilder for å finne det viktigste innholdet og de mest pålitelige og relevante kildene. Informasjonskildene kan være databaser, avdelingsfiler, email, Wikis og dokumentlagre. Når et søk blir utført vil programvaren presentere en liste over dokumenter som er rangert etter relevans. Programvaren sikrer at brukere kun får tilgang til dokumenter de har autoritet til å se.

Spesialiserte informasjonssystem – Artificial Intelligence (AI)

I 1956 foreslo John McCarthy at begrepet Artificial Intelligence (AI) skulle beskrive evnen datamaskiner har til å etterligne eller duplisere funksjoner ved den menneskelige hjernen. På konferansen holdt ved Dartmouth College ble det foreslått at alle aspekter ved læring og andre trekk ved intelligens kan i prinsippet beskrives så presist, at det kan lages maskiner som simulerer dette. Målet ved å bygge AI system er ikke å erstatte menneskelig beslutningstaking, men å gjenskape det for bestemte typer veldefinerte problem.

Watson er en superdatamaskin utviklet av IBM, som var den første kommersielt tilgjengelige datamaskinen som kunne prosessere informasjon som et menneske. Systemet leveres gjennom skyen og kan analysere store mengder data, forstå komplekse spørsmål stilt i naturlig språk og foreslå bevisbaserte svar. Watson kan lære ved å analysere tidligere interaksjoner, få opplæring fra brukerne eller ved å bli presentert for ny informasjon. En tidlig versjon av Watson deltok i game showet Jeopardy, der den slo tidligere vinnere. Datamaskinen kan prosessere menneskelig tale, søke enorme databaser for mulige svar og deretter respondere med menneskelig stemme. Nå blir et skybasert Watson system brukt av doktorer for å utvikle behandlingsprosesser for en rekke sykdommer. Systemet korrelerer data fra pasientens DNA med de siste funnene i medisinske journaler, nye studier, medisinske bilder og kliniske register, for å utvikle en skreddersydd behandlingsprosess til pasienten.



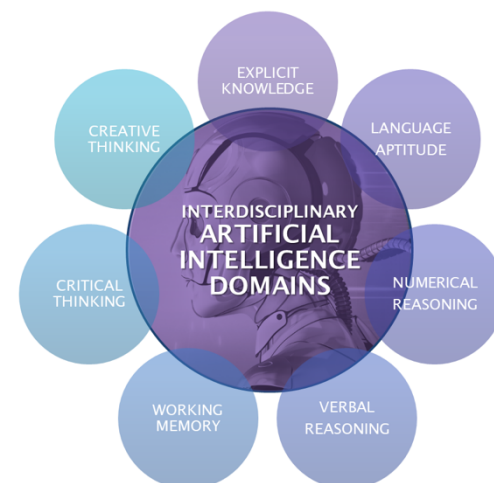
AI i perspektiv

Datamaskiner er flinkere enn mennesker til å overføre informasjon, utføre en rekke beregninger raskt og nøyaktig og utføre komplekse beregninger, men mennesker er flinkere til alle andre aspekter ved intelligens. For eksempel har datamaskiner problemer med å gjenkjenne mønster, tilpasse seg nye situasjoner og trekke konklusjoner når de blir gitt ufullstendig informasjon. AI systemer ser på disse problemene.

Artificial Intelligence systemer omfatter menneskene, prosedyrene, maskinvaren, programvaren, dataen og kunnskapen som trengs for å utvikle datamaskinsystem og maskiner som kan simulere menneskelige intelligente prosesser, slik som:

- **Læring** = anskaffelse av ny informasjon
- **Resonnering** = bruk av logikk for å nå konklusjon
- **Selvkorrigering** = bruke utfallet fra et scenario for forbedre ytelsen til fremtidige scenarioer

AI er et kompleks og interdisiplinært felt som involverer biologi, datavitenskap, matematikk, lingvistikk, nevrovitenskap, filosofi og psykologi. I dag blir AI brukt i mange industrier og applikasjoner.



Intelligensens natur

Fra starten av har fokuset innenfor AI vært å utviklet maskiner som har **intelligent oppførsel**, noe som inkluderer evnen til å:

- **Lære fra erfaringer og bruke denne kunnskapen i nye erfaringer (prøving og feiling)**
- **Håndtere komplekse situasjoner**
- **Løse problemer når viktig informasjon mangler (håndtere usikkerhet)**
- **Bestemme hva som er viktig**
- **Reagere raskt og korrekt i nye situasjoner**
- **Forstå visuelle bilder (perseptivt system)**
- **Prosessere og manipulere symboler**
- **Være kreativ, fantasifull og innovativ**
- **Bruke heuristikk (tommelfingerregel)**

Turing test brukes for å sjekke om en datamaskin har Artificial Intelligence. En menneskelig forhører stiller noen spørsmål og dersom hen ikke klarer å avgjøre om svaret kommer fra en person eller en datamaskin, vil datamaskinen ha AI.

Brain-Computer Interface (BCI)

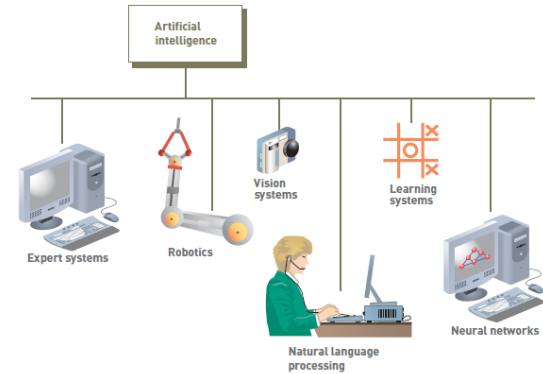
Ideen bak Brain-Computer Interface (BCI) er å koble den menneskelige hjernen direkte til en datamaskin, slik at menneskelige tanker kan kontrollere aktivitetene til datamaskinen.



Realisering av BCI eksperimentet vil la mennesker kontrollere datamaskiner og kunstige kroppsdelene vha bare tanker. Honda Motors har utviklet et BCI system som lar en person utføre bestemte operasjoner, slik som å bøye en fot, med 90% nøyaktighet. Dette systemet bruker en spesiell hjelm som kan måle og sende hjerneaktivitet til en datamaskin (se figur).

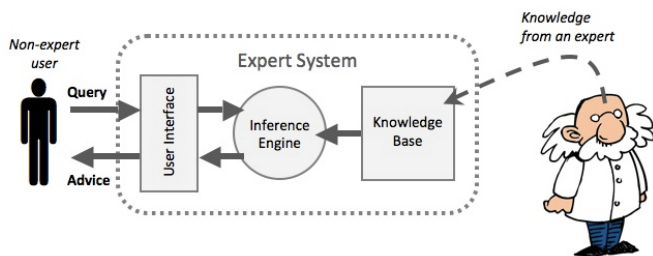
Hovedkomponenter ved AI

AI er et bredt felt som inkluderer flere hovedkomponenter (se figur), som vi nå skal se nærmere på.



Ekspertsystem

Et ekspertsystem består av maskinvare og programvare som lagrer kunnskap og gjør slutninger (inferenser), slik som en menneskelig ekspert. Datastyrt ekspertsystemer bruker

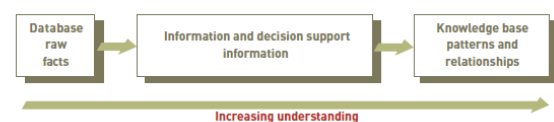
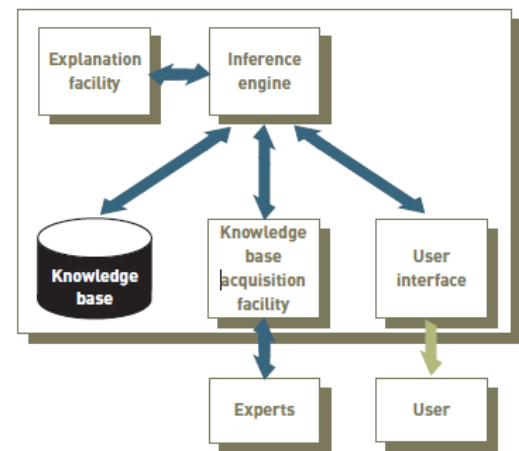


heuristikker, eller tommelfingerregler, for å finne konklusjoner eller komme med forslag. Utviklingen av ekspertsystemer er komplisert, kostbart og tidskrevende, så de blir kun brukt når det er et høyt forventet utbytte, når de kan signifikant redusere risikoer eller når organisasjonen ønsker å fange og ivareta verdifull

menneskelig ekspertise.

Et ekspertsystem består av en samling med integrerte og relaterte komponenter (se figur). Brukeren interagerer med brukergrensesnittet som vil interagere med inferensmotoren, som igjen vil interagere med de andre komponentene i ekspertsystemet for å gi ekspertise. Komponentene er:

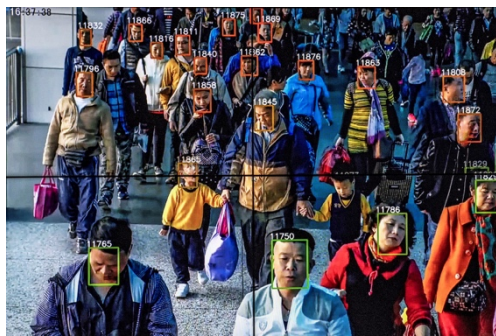
- **Kunnskapsbase** = lagrer all relevant informasjon, data, regler, caser og relasjoner som ekspertsystemet bruker. Det er en naturlig utvidelse av en database og informasjon og decision support system, siden det har økt forståelse sammenlignet med disse. For hvert ekspertsystem må det lages en kunnskapsbase ved å formulere **regler**, som er betinget påstander som kobler betingelser til handlinger eller utfall (kan lagres som IF-THEN påstander).
- **Inferensmotor** = søker etter informasjon og relasjoner i kunnskapsbasen og gir svar, prediksjoner og forslag slik en menneskelig ekspert ville ha gjort. Det er komponenten som gir ekspertrådet. Det finner konklusjonen ved å bruke data gitt av brukeren og reglene i kunnskapsbasen.
- **Forklaringsanlegg (Explanation facility)** = lar brukeren eller beslutningstakeren forstå hvordan ekspertsystemet ankom en bestemt konklusjon eller resultat. For eksempel kan et medisinsk ekspertsystem la doktoren se logikken den brukte for å komme frem til en diagnose. Forklaringsanlegget gir alle faktaene og reglene den brukte for å nå konklusjonen, slik at man kan bestemme om ekspertsystemet behandler dataen og informasjonen på en korrekt og logisk måte.



- **Anskaffelsesplan for kunnskapsbase** = gir en nyttig og effektiv måte å fange og lagre alle komponentene i kunnskapsbasen. Gir brukere og beslutningstakere mulighet til å lage og oppdatere kunnskapsbasen på en brukervennlig måte. Det fungerer som grensesnittet mellom ekspertene og kunnskapsbasen.
- **Brukergrænse** = gjør at det blir lettere for brukere og beslutningstakere å utvikle eller bruke ekspertsystemet.

Robotikk

Robotikk involverer utvikling og produksjon av mekaniske enheter som kan utføre oppgaver som krever høy presisjon eller er kjedelige og/eller farlige for mennesker å utføre. I dag bruker organisasjoner roboter for å utføre kjedelige, skitne og/eller farlige jobber. De blir brukt til å blant annet løfte og flytte tunge paller i varehus, utføre sveiseoperasjoner og observere radioaktivt forurenset områder. Bruksområdene blir stadig utvidet, for eksempel blir roboter brukt for å utføre medisinske operasjoner, assistere og beskytte soldater, hjelpe barn å lære programmering og kreativ problemløsning. Noen frykter at roboter vil stjele jobber fra mennesker, for eksempel kan autonome kjøretøy erstatte lastebilsjåfører, taxisjåfører, osv.

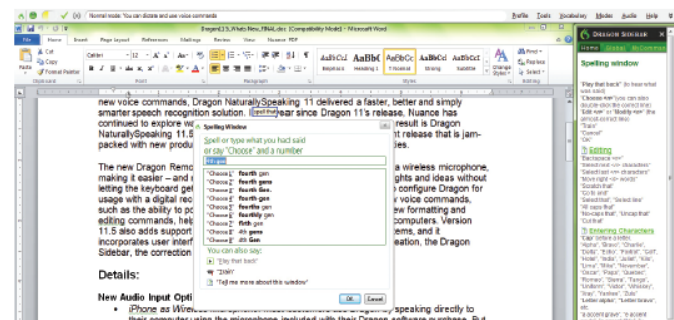


Visjonssystem

Visjonssystem er maskinvare og programvare som lar datamaskiner fange, lagre og prosessere visuelle bilder. 3D-maksin visjonssystem brukes for å øke nøyaktigheten og hastigheten ved industriell inspeksjon av deler. Automatiserte fruktplukkende maskiner bruker vakuum-plukker og visjonssystem for å plukke frukt. Facebook holder på å utvikle et AI visjonssystem, kalt DeepFace, som skaper 3D modeller av ansikter i bilder, noe som kan brukes for å identifisere mennesker basert på ansiktstrekk (97.25% nøyaktighet).

Natural Language prosessering

Natural Language prosessering lar datamaskiner forstå, analysere, manipulere og generere «naturlig» språk, slik som engelsk. Mange selskap gir naturlig language prosessering over telefonen, der man blir gitt en rekke alternativ og kan si hvilket alternativ man velger. Et eksempel er NaturallySpeaking applikasjonen, som bruker kontinuerlig stemmegjenkjenning for å la brukeren snakke til datamaskinen i normal hastighet uten pause mellom ordene. Ved stemmegjenkjenning vil lydbølgene omdannes til ord, og deretter kan systemet reagere på ordene ved å utføre en rekke oppgaver (eks: sette nye alarm på Apple Watch).



Lærende system

Lærende system er en kombinasjon av programvare og maskinvare som lar datamaskinen endre hvordan den fungerer eller hvordan den reagerer på situasjoner basert på feedback den mottar. For eksempel kan man lage datastyrte videospill som kan lære evner ved å prøve og feile. Hvis datamaskinen ikke vinner spillet vil den huske å ikke utføre samme bevegelsene under samme betingelsene igjen. Programvare for lærende system trenger feedback på resultatet til handlinger eller avgjørelser. Denne feedbacken må i minste fall gi

om resultatet er ønskelig eller ikke, og den brukes for å endre hva systemet gjør i fremtiden. Et eksempel er Google som kombinerte lærende system med natural language prosessering i Android telefoner for å redusere antall feil ved ordgjenkjennelse med 25% .

Nevrale nettverk

Nevrale nettverk er datasystem som kan gjenkjenne og handle på mønstre eller trender som den oppdager i store datasett. Det bruker et enormt antall prosessorer som er plassert i en gitterlignende struktur som etterligner hjernestrukturen. Dette gjør at nevralt nettverk kan behandle mye data samtidig og lære å gjenkjenne mønstre. AI Trilogi er et nevralt nettverksprogram som kan kjøre på en standard PC og kan brukes for å lage prediksjoner, hjelpe til i klassifisering og beslutningstaking, osv. For eksempel kan programmet brukes for å detektere kredittkort transaksjoner som sannsynligvis er svindel.

Andre bruksområder for AI

Andre bruksområder for AI inkluderer **genetiske algoritmer** som bruker evolusjonsteorien og konseptet «*survival of the fittest*» for å løse problemer, og **intelligente agenter** som består av programmer og en kunnskapsbase og utfører bestemte oppgaver for en person, en prosess eller et annet program.

Multimedia og virtuell realitet (ikke i F)

Systemer innenfor multimedia og virtuell realitet kan omforme grensesnittet mellom mennesker og informasjonsteknologien ved å tilby nye måter å kommunisere informasjon, visualisere prosesser og uttrykke ideer.

Multimedia er innhold som bruker mer enn en form for kommunikasjon (eks: tekst, grafikk, video, animasjon og lyd), og det kan brukes av organisasjoner for å effektivt nå deres mål. Multimedia verktøy kan hjelpe organisasjonen med å nå deres mål gjennom produksjonen av overbevisende brosjyrer, presentasjoner, rapporter og dokumenter. De fleste organisasjonene bruker ikke alle evnen til multimedia, men holder seg til tekst og grafikkfunksjoner. Presentasjonsprogramvare, slik som Microsoft PowerPoint, kan brukes for å lage presentasjoner med lyd og animasjon som kan vises på en stor skjerm.



Virtuell realitet (VR) kan brukes for å la en eller flere brukere flytte og reagere i et datasimulert miljø. VR simulering krever spesielle grensesnittenheter som overfører severdighetene, lydene og sensasjonene fra den simulerte verden til brukeren. Disse enhetene kan også ta opp og sende deltagerens tale og bevegelse til simuleringsprogrammet. Dermed kan brukere sanse og manipulere virtuelle objekter på samme måte som de ville ha gjort med virkelige objekter. Dette gir deltagerne følelsen av å

How to Design a Neural Network

Neural networks pass an input, like an image, through multiple layers of digital neurons. Each layer reveals additional features of the input. Mathematicians are revealing how a network's architecture — how many neurons and layers it has and how they're connected — determines the kinds of tasks that the neural network will be good at.

When data is fed into a network, each artificial neuron that fires (labeled "1") transmits signals to certain neurons in the next layer, which are likely to fire if multiple signals are received. This process reveals abstract information about the input.

A SHALLOW NETWORK has few layers but many neurons per layer. These "expressive" networks are computationally intensive.

A DEEP NETWORK has many layers and relatively few neurons per layer. It can achieve high levels of abstraction using relatively few neurons.

være en del av den simulerte verdenen. Virtuell realitet kan også referere til applikasjoner som ikke er altoppslukende, for eksempel mus-kontrollert navigering gjennom tredimensjonale omgivelser på en grafisk monitor. Noen VR-applikasjoner lar brukeren se virkelige omgivelser med overlagte virtuelle objekter. Augmentert realitet (AR) er en nyere form for VR der digital data kan legges over ekte bilder. Applikasjoner med VR brukes innenfor blant annet medisin, utdanning, trening, eiendomsmegling, turisme og underholdning.



Andre spesialiserte system

Spesialiserte system kan hjelpe organisasjoner og individer til å oppnå deres mål. I det siste har det dukket opp en rekke spesialiserte system som hjelper organisasjoner og individer på nye og spennende måter. Eksempler er:

- **Hjelpemiddelsystemer (*assistive systems*)** = et bredt spekter med hjelpende, tilpasningsdyktige og rehabiliterende enheter som hjelper mennesker med nedsatt funksjonsevne til å utføre oppgaver som de tidligere ikke var i stand til eller hadde store problemer med å utføre
- **Spillteori (*game theory*)** = en matematisk teori som hjelper til med å utvikle strategier for å maksimere gevinster og minimere tap, samtidig som man overholder et gitt sett med regler og begrensninger
- **Informatikk** = kombinasjonen av informasjonsteknologi og tradisjonelle fagdisipliner som medisin eller vitenskap, som er fokusert på innvirkningen på individer, organisasjoner og samfunn. Det representerer skjæringspunktet mellom mennesker, informasjon og teknologi.

BPMN modellering

Denne delen av kompendiet er basert på boka *BPMN Method & Style* av Bruce Silver.

Kapittel 1 – Dårlig BPMN, god BPMN

BPMN (Business Process Model and Notation) er ledende standard for modellering i BPM (Business Process Modeling/Business Process Management). Første versjon ble gitt i 2004, OMG standarden ble gitt i 2006 (standard = ikke eiet eller kontrollert av noen) og versjon 2.0 ble gitt i 2011. Det finnes flere ulike som kan brukes for å lage BPMN modeller, men ingen inkluderer alle deler av språket og semantikken er definert av standarden. **BPMN er et uttrykksfullt språk som bruker diagrammer for å kompakt beskrive trekk ved business prosesser.** Det kan også gi tekniske detaljer som kontrollerer utføringen av prosesser. BPMN fungerer som en kobling mellom business og IT.

BPMN appellerer til forretningsmennesker fordi det ligner på tradisjonelle flytdiagram, men i dybden har det flere unike egenskaper:

1. BPMN har en **formell spesifisering av former og symboler**. Det har regler som kontrollerer hvordan formene kan brukes og kobles sammen
2. BPMN kan beskrive **hendelsesutløst oppførsel**
3. BPMN kan beskrive **kommunikasjon mellom prosesser og eksterne enheter**, slik som kunder, leverandører, osv.

BPMN er et rikt språk, noe som er nødvendig for å beskrive komplekse ideer. Man trenger opplæring og verktøy for å kunne bruke det riktig og effektivt. **Business prosess modellering (BPM) er mer enn BPMN**, for eksempel vil det involvere å beskrive forhold til konkurrenter og leverandører, markedet, value chain, IT system, osv. Disse elementene kan beskrives av separate modeller som kan kobles til BPMN modellen. Det vil være umulig å lage en modell som beskriver alle aspektene ved BPM. **En av hovedgrunnene til at BPMN er akseptert som en standard, er at den ikke forsøker å gjøre for mye.**

Metoder og stil

En gode BPMN modell er:

- **Korrekt** = diagrammet følger reglene som gis av BPMN spesifikasjonen
- **Tydlig** = logikken til prosessen er entydig og åpenbart forklart i diagrammet. Modellen skal ikke avhenge av dokumentasjon ved siden av. Prosesslogikken beskriver hvordan man går fra en oppgave til neste.
- **Fullstendig** = diagrammet gir flyten av aktivitet, hvordan prosessen starter, signifikante slutttilstander og kommunikasjon med eksterne entiteter
- **Konsistent** = hvis de gis samme fakta om prosesslogikken, skal det kunne lages flere modeller som er tilnærmet like.

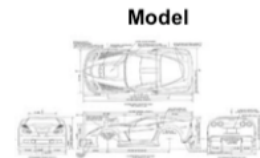
BPMN modeller med disse egenskapene følger Method and style prinsippet.

Kapittel 2 – Hva betyr en modell?

Modeller kan brukes for å fremheve essensielle deler av realiteten, og det er raskere og billigere å lage modeller sammenlignet med å lage fullstendige artefakter. Modellene er like den fullstendige artefaktet, slik at de kan brukes for å oppdage konsekvenser ved avgjørelser, for eksempel ved analyse, simulering, inspeksjon, osv. Modeller kan brukes i risikofri testing, slik som *what-if* scenarier eller ulykkesimulering. De er viktige for å produsere med innovative resultat med bedre kvalitet. Modeller kan ses på som en abstraksjon, projeksjon eller dekomponering av den virkelige verden.



Vi skiller mellom objektet som eksisterer i den virkelige verden og modellen som brukes for å fremheve essensielle deler av objektet. Modeller kan brukes for å bedre kommunikasjonen, sikre kvaliteten, analysere designet, osv.



En prosessmodell brukes for å beskrive logikken til aktivitetsflyten fra start til slutt, dvs. stegene langs alle banene fra starttilstanden til alle mulige endetilstander. Fra diagrammet skal prosesslogikken være klar og forståelig for business personer, og semantisk presis slik at den kan brukes av en utvikler.

Hva er en aktivitet?

En aktivitet i BPMN er en handling som blir utført flere ganger i virksomheten, altså en arbeidsenhet som gjentas. Hver instans av aktiviteten representerer samme handling på et annet arbeid. BPMN aktiviteter er diskrete med veldefinert start og slutt. Når en instans er slutt er den fullstendig over.

Hva er en prosess?

En prosess er en sekvens av aktiviteter som fører fra en initial tilstand til en definert slutttilstand. Starten til en prosess er markert av en utløsende hendelse, slik som en forespørsel eller bestilling. En BPMN prosess er diskret, blir utført gjentatte ganger i virksomheten og har en veldefinert start og slutt. Det er viktig at hver aktivitet i sekvensen er diskret, utføres gjentatte ganger og har veldefinert start og slutt. **En prosessmodell er et kart over alle mulige baner fra starttilstanden til enhver definert slutttilstand.** **Prosesslogikken definerer alle mulige aktivitetssekvenser fra prosessens utløsende hendelse til én av dens slutttilstander.** Prosessmodellen viser altså all prosesslogikk fra start til slutt. Alle instanser ved en prosess må følge en bane i prosessmodellen. Hvilken bane den tar vil avgjøres av informasjonen instansen har akkumulert (eks: mottatte meldinger, data produsert i aktiviteter, osv.)

Business prosesser

Business prosesser består av et sett med aktiviteter som utføres i en organisasjon. Disse aktivitetene blir koordinerte for å realisere målet til organisasjonen. **Hver business prosess blir utført av en enkel organisasjon, men den kan interagere med business prosesser som utføres av andre organisasjoner.** Modellering av business prosesser er en sentral aktivitet i informasjonssystemer og det brukes for å forstå, vurdere og støtte prosessene. Modellene kan fange mange høy-nivå krav og begrensninger ved systemet

Hva kan BPMN modeller svare på?

En BPM modell vil gi rekkefølgen til aktiviteter, når de skjer og under hvilke betingelser. Den beskriver hva som skjer etter en aktivitet er ferdig, men gir lite om hva som skjer innenfor aktiviteten. Den beskriver ikke hvordan, hvorfor eller hvor en aktivitet utføres. Merkelapper på aktivitetene er det eneste som kan gi antydning til hva aktiviteten er og hvem den utføres av. Dette er ikke en del av prosesslogikken og blir derfor utenfor domenet til BPMN.

Tre nivåer for bruk

De som bruker BPMN er interessert i ulike nivåer med prosessdetaljer. BPMN modellering kan deles inn i tre nivåer:

- **Nivå 1: Beskrivende modellering** = brukes for å enkelt beskrive prosessflyten ved å bruke former og symboler fra tradisjonelle flytdiagram. Fokuset i dette emnet.

- **Nivå 2: Analytisk modellering** = er mer nøyaktig mht. unntak og hendelser. Støtter kvalitativ og kvantitativ analyse
- **Nivå 3: utførende modellering** = grafiske modeller som kan omformes til XML-baserte spesifikasjoner som kan kjøres i prosessmotorer

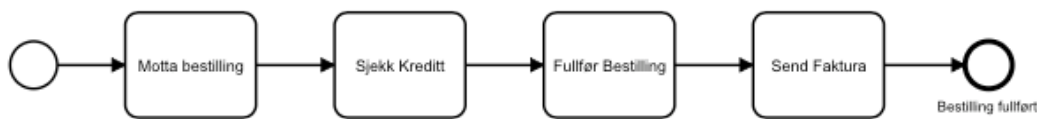
Ved å dele opp BPMN i nivåer, gjør det at modellutveksling blir mulig.

Kapittel 3 – BPMN via eksempler

Vi ser på bruk av BPMN ved nivå 1, altså beskrivende modellering.

En enkel bestillingsprosess

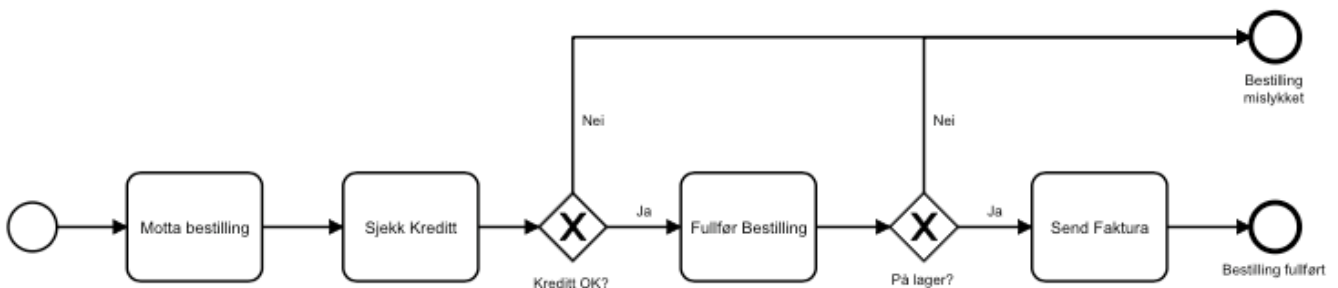
Vi ser på prosessen av å håndtere en enkel bestilling. Selskapet vil motta bestillingen, sjekke kreditten til kjøperen, fullføre bestillingen og sende en faktura. Figuren viser hvordan denne prosessen ser ut i BPMN. **Dette er ikke en prosessmodell, men kalles en happy path, siden det er en normal sekvens med aktiviteter uten unntak.**



Den tynne sirkelen ved begynnelsen kalles en **starthendelse**, mens den tykke sirkelen ved enden kalles en **slutthendelse** og gir at prosessen er fullført. Rektanglene representerer **aktiviteter på formen Verb-Substantiv**.

Unntak og distinkte endetilstander

En mer fullstendig modell av prosessen vil involvere unntaksbaner:



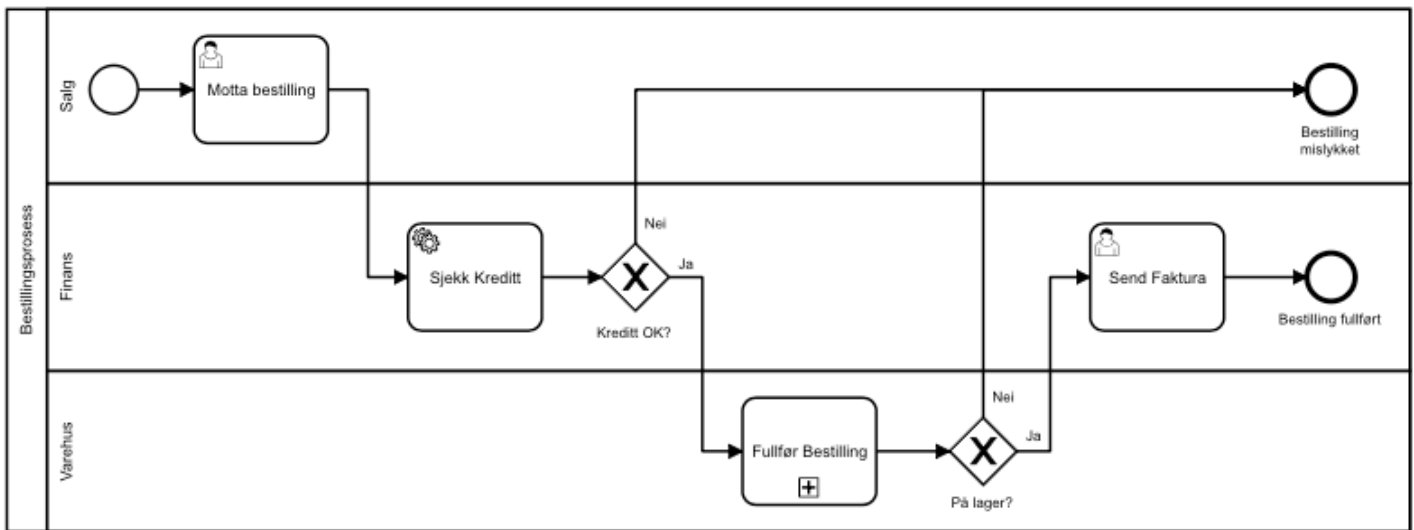
Diamantformene kalles gateways (porter) og representerer forgreininger i prosessflyten.

Det finnes flere typer *gateways* og de på figuren som har et kryss i seg (eller er tom) kalles **XOR gateways og betyr at en av banene skal tas basert på en betingelse**. Sekvensflytene ut av en gateway kalles *gates*. **Gateways brukes for å skille unntaksbaner fra happy path**. Legg merke til at vi har to slutttilstander. Method and Style prinsippet krever at man bruker separate endehendelser for å indikere distinkte endetilstander, slik som suksess og fiasko, og disse må gis navn. Legg også merke til at diagrammet nå beskriver tre distinkte baner fra start til slutt.

Lanes, Pools og aktivitetstyper

Vi bruker lanes for å indikere hvem som utfører hver aktivitet i prosessen. Flere lanes utgjør til sammen en **pool, som er rektangelet som inneholder hele prosessen**. En pool bør merkes med navnet til hele prosessen. **Vi bruker oppgavetype ikoner for å skille mellom ulike typer aktiviteter.** For eksempel er det nyttig å skille mellom **User Task** som utføres av mennesker (menneskeikon) og **Service Task** som utføres automatisk (tannhjul). Lanes brukes

egentlig kun for *User tasks*, men kan også brukes for *Service tasks*. Gateways og hendelser plasseres i lanes etter hva som er praktisk.

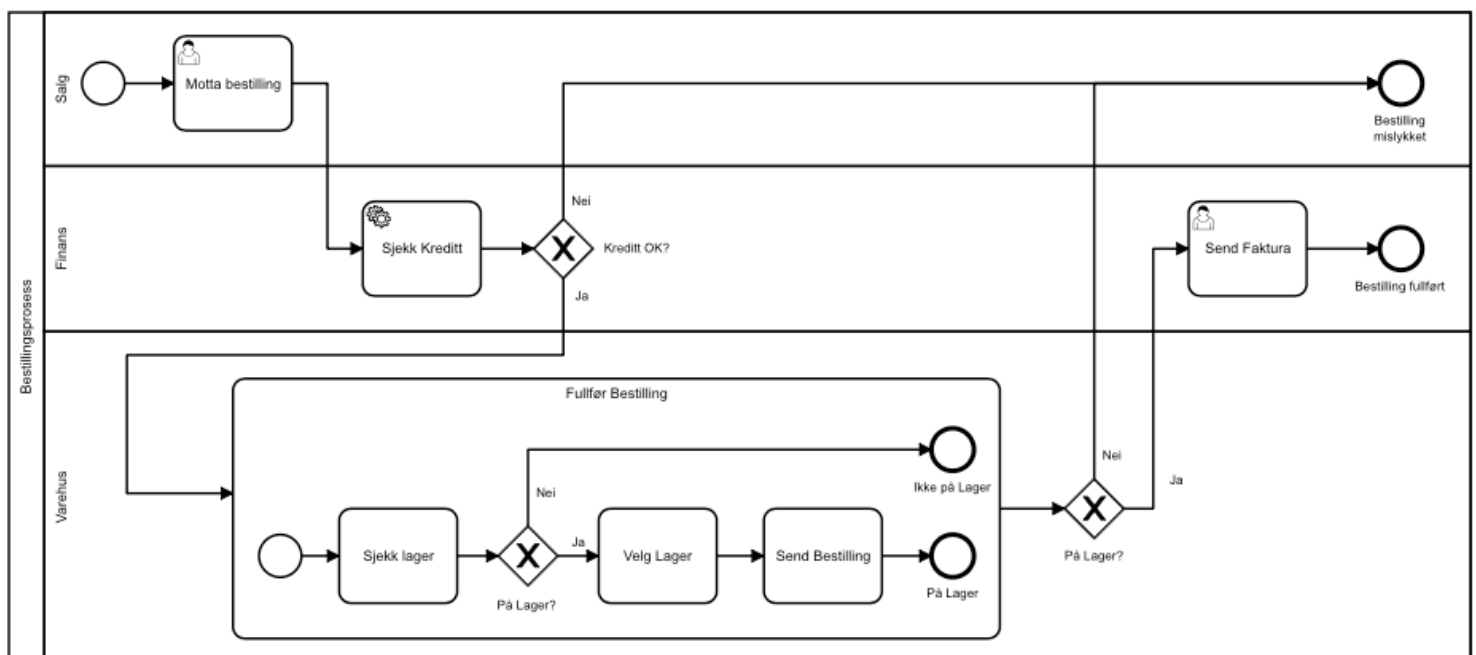


Subprosesser – kollapset og utvidet

På figuren over kan vi se at Fullfør Bestilling har et lite [+] ikon, som representerer at det er en **kollapset subprosess**. En subprosess er en aktivitet som inneholder deler som kan uttrykkes som en prosessflyt. En oppgave er en aktivitet som ikke består av definerte deler. Subprosessen er en aktivitet (et steg i prosessen som utfører arbeid), samtidig som det er en prosess (en flyt av aktiviteter fra starttilstand til en eller flere slutttilstander). Subprosesser kan representeres som:

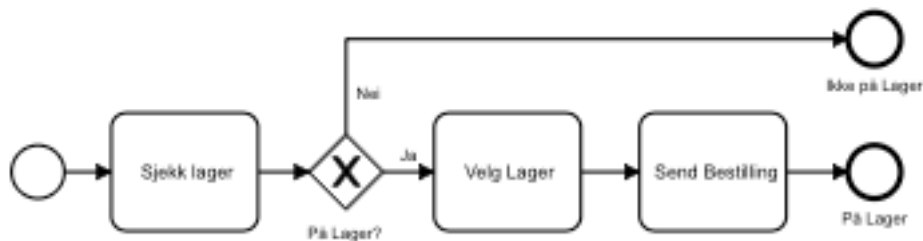
- **Kollapset subprosesser** = en enkel aktivitetsrektangel med [+] ikon
- **Utvidet subprosesser** = en utvidet subprosess form med innhold (rundt rektangel)

Disse betyr det samme, men utvidet subprosess gir flere detaljer. En utvidet subprosess ser ut som en prosess ved at den har en starttilstand, en flyt av aktiviteter og en sluttendelse for hver distinkt endetilstand.



Prosessnivå og hierarkisk stil

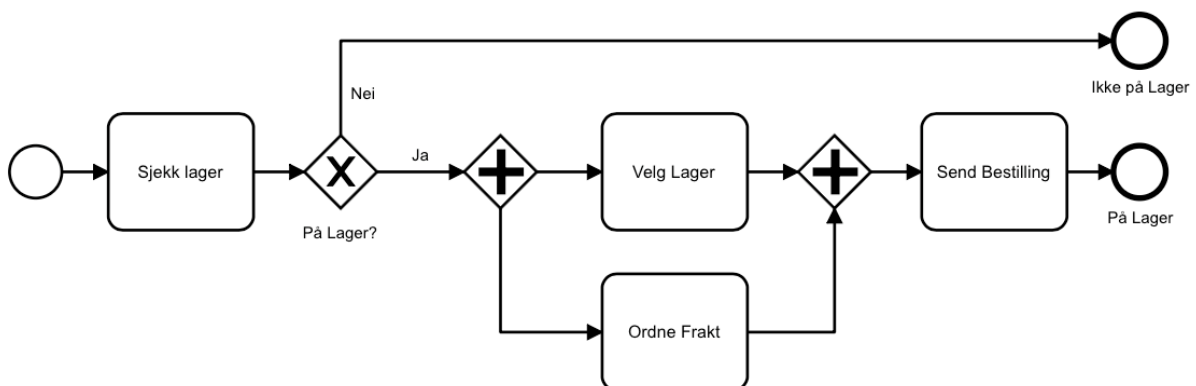
Proessen innenfor en subprosess kalles barnenivå prosess med hensyn til den overordnede foreldrenivå prosessen. En subprosess kan inneholde en subprosess, som kan inneholde en subprosess, osv. Det er ingen grense på antall nivåer man kan nøste subprosesser. Ved utvidete subprosesser vil forelde- og barnenivåene vises i samme diagram, og dette kan bruke mye plass. En annen løsning er å vise barnenivå prosessen i et separat diagram, noe som kalles **hierarkisk utvidelse**, siden det viser ende-til-ende prosessen som et hierarki av diagrammer. På papir blir matchende merkelapper brukt for å koble barnenivå diagrammet med den kollapsede subprosessen i foreldrenivå prosessen. Figuren under viser utvidelsen av Fullfør Bestillingen i et barnenivå diagram:



Legg merke til at det ikke er noen pool eller rektangel rundt prosessen. En barnenivå utvidelse kan ha egne lanes eller arve lanes fra den kollapsede subprosessen i foreldrenivå diagrammet. Utvidet subprosesser kan brukes for enkle diagrammer, mens hierarkisk stil brukes for komplekse prosesser (unngå for mye informasjon på ett ark).

Parallelle splitter og join

Under fullføringen av bestillingen ønsker vi at fraktordningen skal skje samtidig som man velger lager (dvs. henter produkter), altså parallelt. Diagrammet blir følgende:



For å lage parallele flyter bruker vi AND-gateways med plusstegn i diamantformen. På figuren kan vi se at det er to typer AND-gateways:

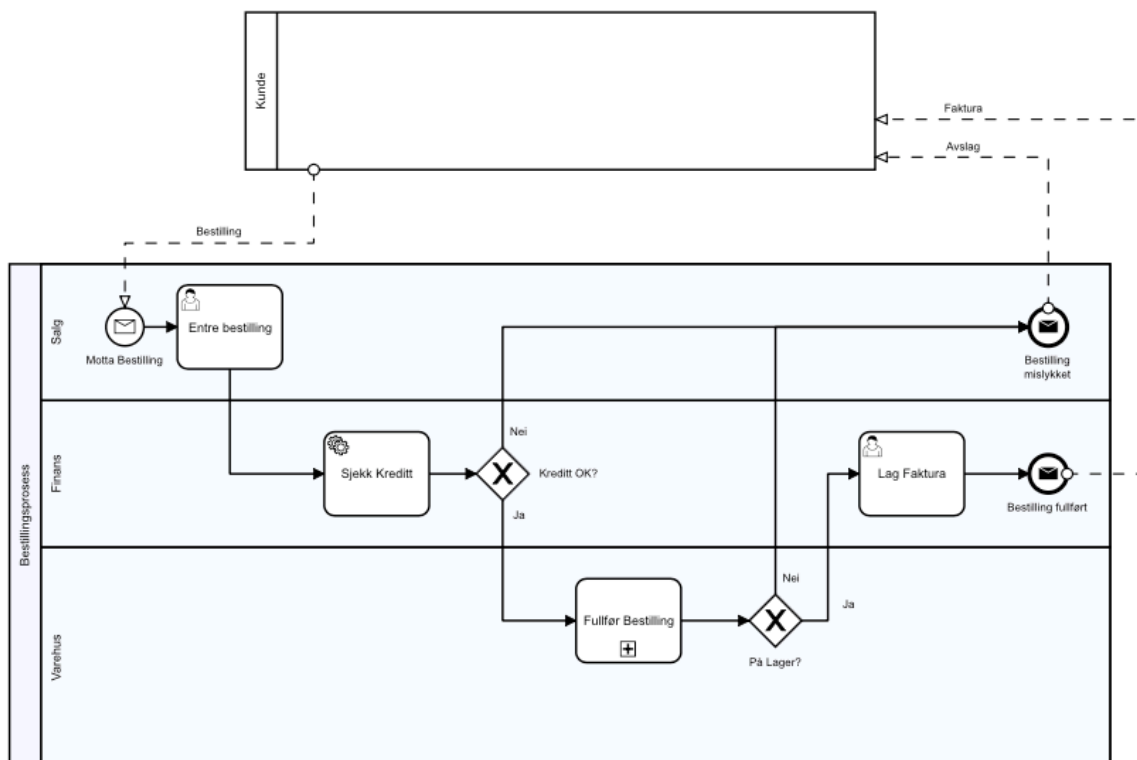
- **AND-splitt (parallell splitt)** = en parallell gateway med én sekvensflyt inn og to eller flere sekvensflyter ut. Flyten deles opp uten betingelse i parallelle segmenter med aktiviteter som skjer samtidig.
- **AND-join (synkroniserende join)** = en parallell gateway med to eller flere sekvensflyter inn og én sekvensflyt ut. Denne gatewayen vil vente på at all inngående flyt har ankommet før den tillater utgående flyt. På figuren betyr dette at bestillingen ikke kan sendes før man har valgt ut varene på lageret OG ordnet frakten

BPMN tillater at man ikke inkluderer AND-join, altså at parallele splitter ikke blir fusjonert. Parallele segmenter kan føre til separate slutthendelser, og i så fall vil ikke prosessen være fullført før alle parallele segmenter har nådd slutthendelsen. En analogi er en elv som splittes i to deler, der vannet vil renne nedover begge delene og ikke bare en.

Samarbeid og Black-Box pools

En kunde av prosessen skal IKKE plasseres som en lane i poolen til prosessen, fordi kunden er ekstern til prosessen og ikke en del av den. For eksempel kan man vi se på situasjonen der man befinner seg på Amazon.com og har plassert varer i handlekurven, men avbryter bestillingen før man er ferdig. I dette tilfellet har man ikke laget en instans av bestillingsprosessen til Amazon. Denne prosessen starter når Amazon mottar en bestilling. **Hvis det er en som forespør en prosess (anmoder) er det som regel best å modellere dette som en ekstern deltager.**

En ekstern entitet blir modellert som et tomt, separat pool i diagrammet. Et slikt pool inneholder ingen flyt, og det kalles et **black-box pool**, siden den interne prosessen er ukjent for oss. Et black-box pool blir merket med navnet til rollen eller entiteten, for eksempel Kunde:



Kunden interagerer med prosessen ved å utveksle meldinger, og i BPMN brukes begrepet **message** for å indikere enhver kommunikasjon mellom prosessen og eksterne deltagere. **Meldingsflyt er mellom to pools og indikeres av en stiplet pil med sirkel ved halen og tomt pilhode, mens sekvensflyt er innenfor ett pool og indikeres av hard pil uten sirkel ved halen og fylt pilhode.** Et diagram som viser interaksjon mellom prosessen og eksterne deltagere via meldingsflyt kalles **samarbeidsdiagram**.

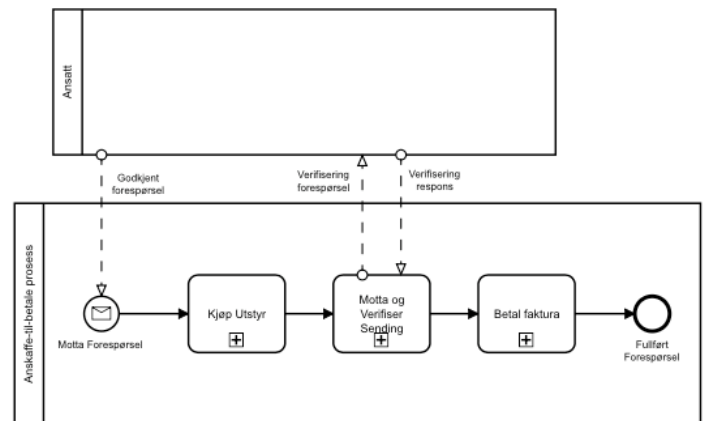
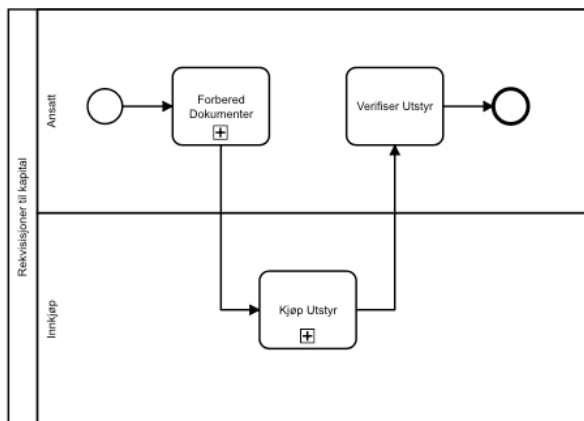
Konvolutt ikonet plasseres innenfor hendelser for å markere at disse hendelsene sender og/eller mottar meldinger. Starthendelsen har en meldingstrigger og sluthendelsen har en meldingsresultat. **Meldingstriggeren er mye brukt og den signaliserer at en instans av prosessen blir laget når den mottar en melding (eks: bestilling).** Hvis en ny bestilling ankommer rett etter den første, vil det lages en ny instans av prosessen. Det er kun toppnivå prosesser som kan ha meldingstriggende starthendelser, så subprosesser har ingen triggerikon. **Meldingsresultatet signaliserer at det vil sendes en melding når hendelsen blir nådd.** Hvide konvolutter betyr at det mottas meldinger, mens svarte konvolutter betyr at det sendes. Siden det er hendelsene som mottar og sender hhv. bestilling og faktura, har vi endret aktivitetene til Endre Bestilling og Lage Faktura. **Vi vil ikke duplisere handlingene til start- og sluthendelsene.**

Man kan ikke plassere noen aktiviteter slik som «Fyll ut bestilling» i en black-box pool, fordi en pool som inneholder flytelementer vil være en prosess pool og må dermed representere en fullstendig prosess fra start til slutt. Hvis eldingsflyter er festet til kanten av et pool, må dette være et black-box pool og ikke et prosess pool. For at vi skal kunne tegne meldingsflyt i denne situasjonen må tegne hele kjøpsprosessen for kunden, og det er ikke sikkert at den er kjent for selgeren.

Starthendelser og prosessinstanser

En meldingstrigget starthendelse indikerer at prosessen starter når den mottar en forespørsel (eks: bestilling, lånesøknad, osv.). Disse starthendelsene bør merkes med **Motta [navn på melding]**, slik som **Motta Bestilling**. De fleste prosessene blir trigget av en forespørsel. Meldingstrigget starthendelse indikerer at prosessen startes av en ekstern forespørsel og poolen (ofte black-box) ved halen til meldingsflyten er den som forespør prosessen.

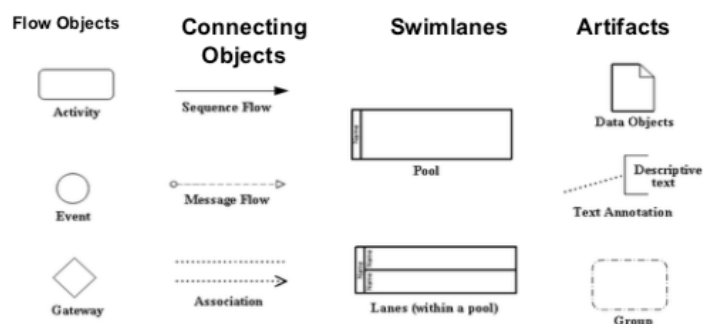
En meldingstrigget starthendelse indikerer også at prosessinstansen er et forsøk på å tilfredsstille en enkel forespørsel. Hver aktivitet i prosessen er relatert til denne forespørselen og de vil ikke si noe om tilstanden til andre forespørsler. Forespørselen blir alltid gitt av en ekstern deltager, selv om det ikke er en kunde. Hvis den som forespør prosessen har definerte oppgaver den skal utføre som en del av prosessen, må vi heller modellere dette som en lane i prosesspoolen og bruke None starthendelse (dvs. ingen trigger). Figurene under viser eksempler på dette. Det er kun når det er en ekstern deltager som forespør prosessen at vi bruker black-box pool, meldingsflyt og meldingstrigget starthendelse.



Kapittel 4 – Nivå 1 palett

Nivå 1 palettet kalles *Descriptive Process Modeling Conformance subclass*. Dette nivået kan brukes for å beskrive nesten alle prosesser, hvis man ikke er så interessert i hendelsesutløst oppførsel. Følgende er alle elementene i nivå 1 palettet:

- **Aktivitet:** oppgaver (*User, Service, None*), subprosesser og *Call Activity*
- **Gateway:** eksklusiv og parallell
- **Starthendelse:** None, Melding, Timer
- **Slutthendelse:** None, Melding, Terminate
- **Sekvensflyt og meldingsflyt**
- **Pool og Lane**
- **Dataobjekt, Datalager og Dataassosiasjon**
- **Dokumentering**
- **Artefakt:** Tekstannotering, Assosiasjon og Gruppe

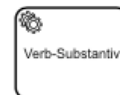
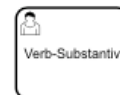


Aktivitet

En aktivitet representerer en arbeidsenhet som utføres i prosessen, og det blir alltid representert som en avrundet rektangel. Det er eneste BPMN element som har en **utfører**. En aktivitet vil enten være en oppgave (atomisk) eller en subprosess (sammensatt).

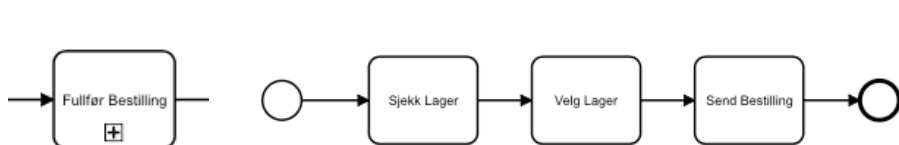
Oppgave

En oppgave er en atomisk aktivitet (dvs. udelelig) og er en enkel handling. Den blir representert som en avrundet rektangel med oppgavetype i øvre, venstre hjørne. Den bør gis navn på formen Verb-Substantiv. De tre vanligste oppgavetyperne er: **User task** (utføres av person), **Service task** (utføres automatisk) og **Abstract task** (typen er udefinert). Figuren til venstre viser andre oppgavetyper.

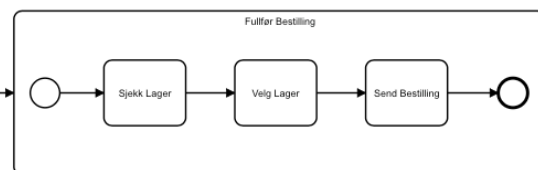


Subprosess

En subprosess er en sammensatt aktivitet (dvs. delelig) og kan beskrives som en barnenivå prosess. Den kan representeres på to ulike måter: (1) kollapset subprosess gis som aktivitet med [+] ikon og subprosessen gis som separat barnenivå diagram (hierarkisk utviding) og (2) utvidet subprosess med rektangel og innhold i prosesspool. Disse metodene har samme semantisk betydning, og den eneste forskjellen er det grafiske utseende. Ved utvidet subprosess er det viktig at sekvensflytene går til kantene av rektangelet og at det inneholder starthendelse og slutthendelse(r). **Starthendelsen i en subprosess har ingen trigger, fordi det er alltid ankomsten av sekvensflyt som trigger subprosesser.**



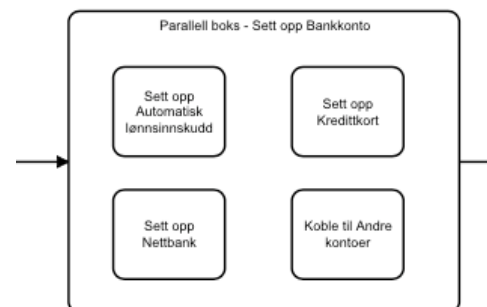
(1) Kollapset subprosess med separat barnenivå diagram (hierarkisk utviding)



(2) Subprosess utviding

Ved den hierarkiske stilen (1) kan top-nivå diagrammet fungere som en ende-til-ende visualisering, fordi ved ett glans på diagrammet kan man se hovedstegene ved prosessen, betydningen til prosessinstansen, mulige endetilstander og interaksjoner med kunder, leverandører og andre prosesser. Ved top-down BPMN modellering begynner man med det øverste nivået og kollapset subprosesser, før man deretter implementerer subprosessene.

En parallell boks er en subprosess som inneholder et sett med aktiviteter som ikke er koblet sammen av sekvensflyt, og det er ingen starthendelse eller slutthendelser. Når denne subprosessen starter, vil alle barneaktivitetene gjennomføres parallelt. De kan fullføres i hvilken som helst rekkefølge, men alle må fullføres før subprosessen kan være ferdig.



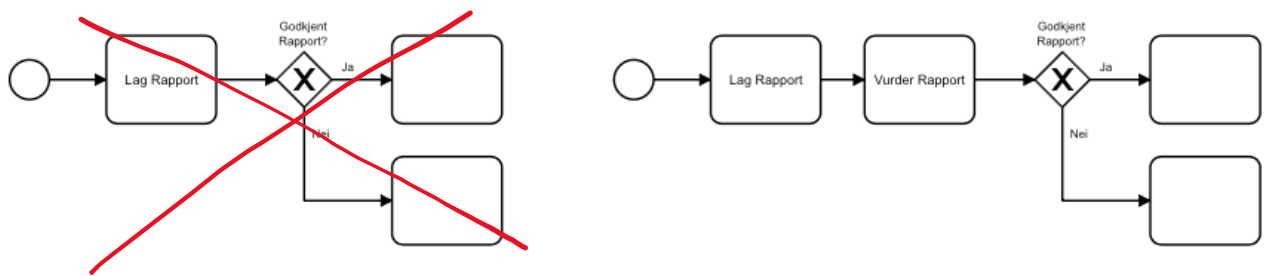
Call Activity

Call Activity er en subprosess som brukes av flere prosesser, slik at det er bedre å definere den uavhengig av prosessene (dvs. utenfor pools) og heller la dem kalle på subprosessen ved behov. I diagrammet blir Call Activities representert som en subprosess med tykke kanter.

Gateway

En gateway er et diamantformet element som kontrollerer prosessflyten ved å dele den opp i alternative baner eller samle sammen diskrete flyter til én flyt. Det er viktig at gatewayen ikke gjør en handling, men bare vurderer resultatet til handlingen foran. For

eksempel diagrammet til venstre er feil, siden gatewayen vurderer rapporten og sjekker om den er godkjent. Diagrammet til høyre er riktig fordi den sjekker bare resultatet til vurderingen i aktiviteten foran.



Eksklusiv databasert gateway (XOR)

Diamantformet element med X eller ingenting inni. Vi skiller mellom to typer XOR-gateways:

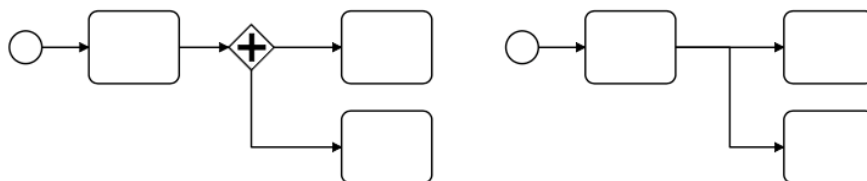
- **XOR-splitt** = eksklusiv betyr at kun én av de utgående sekvensflytene blir aktivert for hver instans (XOR-splitt), og databasert betyr at hvilken flyt aktiveres bestemmes av en betingelse som evaluerer prosessdata. Betingelsen gis av merkelappen på gatewayen. XOR-splitt gateways kan ikke utelates, for da har man ingen måte å gi betingelsen og det blir en parallell splitt.
- **XOR-join** = sekvensflyten som ankommer først er den som blir overført til utgående flyt. Resten blir ignorert. XOR-join gateways kan utelates.

Parallell databasert gateway (AND)

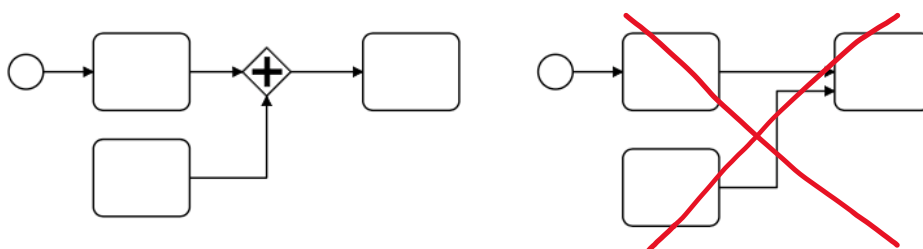
Diamantformet element med + inni. Vi skiller mellom to typer AND-gateways:

- **AND-splitt** = én sekvensflyt inn og to eller flere sekvensflyter ut. All utgående sekvensflyt vil gå parallelt, uten betingelse. Parallelle baner overlapper i tid og kan bli joined eller ende i separate endehendelser (alle parallelle baner må nå enden før prosessen kan være fullført. AND-splitt gateway kan utelates, fordi en aktivitet eller hendelse som har flere utgående sekvensflyter indikerer en parallell splitt
- **AND-join** = to eller flere sekvensflyter inn og én sekvensflyt ut. Den krever at alle innkommende sekvensflyter ankommer før utgående flyt blir aktivert. AND-join kan ikke utelates, fordi flere inngående sekvensflyter indikerer eksklusiv join.

AND-splitt kan utelates, fordi flere utgående sekvensflyter indikerer parallell splitt:



AND-join kan ikke utelates, fordi flere inngående sekvensflyter betyr at aktiviteten gjennomføres flere ganger (hvis det er en tidligere XOR-splitt vil dette være XOR-join).



Flere utgående sekvensflyter er parallell splitt, mens flere inngående sekvensflyter er eksklusiv join. Det er generelt greit å bruke gateways for da slipper man å gjøre feil!

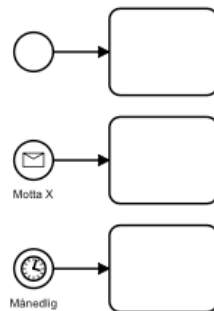
Hendelse

En hendelse representerer noe som skjer som påvirker prosessen ved at det påvirker prosessflyten. Basert på hvor den skjer kan hendelsen være en start-, mellom- eller slutthendelse.

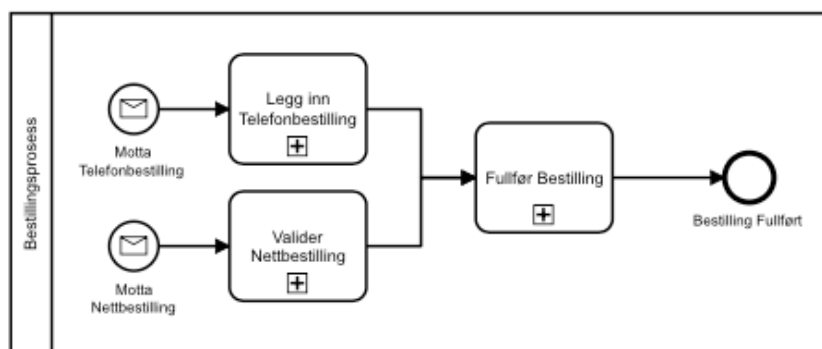
Starthendelse

Starthendelsen indikerer hvor prosessen eller subprosessen starter, og den blir alltid gitt som en sirkel med tynne kanter. Som regel er det kun én starthendelse, men prosessen kan ha flere. **I topp-nivå prosesser kan ikoner brukes innenfor sirkelen for å indikere hva som trigger (initialiserer) prosessen.** En subprosess kan ikke ha triggerikoner, siden den blir alltid trigget av innkommende sekvensflyt. Ulike typer starthendelser er:

- **None starthendelse (ingen ikon)** = ingen trigger, noe som betyr at prosessen initieres av uspesifisert trigger eller manuelt av en oppgaveutfører. Dette brukes alltid i subprosesser
- **Message starthendelse (hvit konvolutt)** = prosessen initieres når den mottar en melding, som er et signal som kommer utenfra prosessen. Prosessen starter som følge av en ekstern forespørsel og prosessinstansen er håndtering av denne forespørselen. Markerer med «Motta X», der X er navnet på meldingen. Man må også inkludere meldingsflyten som merkes med navnet på meldingen.
- **Timer starthendelse (klokke)** = prosessen er planlagt og ofte gjentakende. Den bør merkes for å gi planen (eks: månedlig eller Fredag 16.00). Prosessinstansen er én gjennomføring av den planlagte prosessen og ulike instanser er uavhengige av hverandre.
- **Multiple starthendelse (pentagon)** = prosessen kan initieres av flere ulike triggere, for eksempel melding, timer eller none. Banen ut av starthendelsen er den samme uansett hvilken trigger som mottas. Merkelappen bør indikere alle mulige triggere



Når den initiale prosessaktiviteten avhenger av hvilken trigger som skjer, kan man bruke flere starthendelser. Dette kan bare gjøres i toppnivå diagrammer. Hver starthendelse representerer en alternativ trigger for prosessen. **Når prosessen er initialisert vil den ignorere signaler som andre starthendelser mottar.** Disse signalene vil i stedet initialisere nye prosesser. Dette brukes ved kanalavhengig start, for eksempel når prosessen trigges av ulike typer kundeforespørsler:



Multiple starthendelse brukes når flere triggere initierer samme prosessaktivitet, mens alternative starthendelser brukes når flere triggere initierer ulik prosessaktivitet.

Slutthendelse

Slutthendelsen indikerer hvor prosessen eller subprosessen slutter, og den blir alltid gitt som en sirkel med tykke kanter. Det er vanlig at prosessen eller subprosessen inneholder flere slutthendelser. Slutthendelsen kan tegnes med et fylt ikon som indikerer hva som skjer når slutten nås.

Ulike typer slutthendelser er:

- **None slutthendelse (ingen ikon)** = ingenting skjer. De representerer ikke distinkte endetilstander, så hvis alle parallelle banene ender i None slutthendelser impliserer dette at de blir joined.
- **Message slutthendelse (svart konvolutt)** = en melding blir sendt når slutthendelsen nås. En meldingsflyt bør tegnes fra hendelsen til det eksterne pool.
- **Terminate slutthendelse (svart prikk)** = når denne nås vil prosessen eller subprosessen umiddelbart termineres, uansett om noen parallelle baner ikke har nådd slutthendelsen enda. Kan modellere unntak i en parallell bane.
- **Multiple slutthendelse (svart pentagon)** = flere hendelser utløses, for eksempel kan det sendes flere meldinger (representeres av flere utgående meldingsflyter)

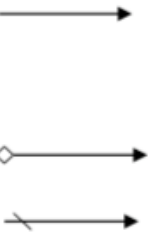


Mellomhendelse

En mellomhendelse skjer etter starthendelsen og før slutthendelser. Betydningen til mellomhendelsen vil avhenge av hvilket ikon den har inni sirkelen, om det brukes to ringer og plasseringen i diagrammet. Hvis hendelsen har et svart ikon betyr det at den genererer et triggersignal, mens hvis den har et hvitt ikon betyr det at den venter på triggersignalet. Det finnes mellomhendelser som ikke støtter disse to typene oppførsel.

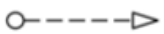
Sekvensflyt

Sekvensflyten representerer den sekvensielle utføringen av steg i prosessen, og den gis som en solid pil med fylt pilhode. Når noden ved halen er fullført, kan noden ved hodet begynne. Disse nodene er aktiviteter, gateways eller hendelser. **Alle aktiviteter, gateways og hendelser i et prosessnivå må ligge på en kontinuerlig kjede av sekvensflyt fra starthendelsen til slutthendelsen.** De kan ikke krysse en subprosess eller et pool. Vi kan skille mellom betinget flyt (diamant på enden) og default flyt (skrå linje)



Meldingsflyt

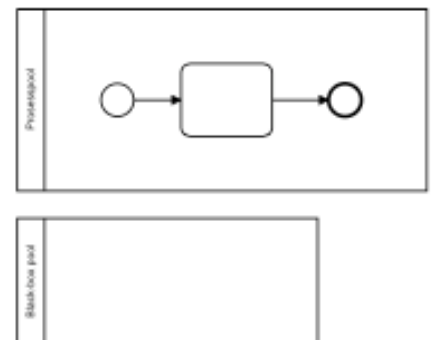
Meldingsflyten representerer en kommunikasjon mellom prosessen og en ekstern entitet, og den gis som en stiplet line, med sirkel ved halen og tomt pilhode. Den kan være koblet til aktiviteter, message eller multiple hendelser og black-box pools. Den kan ikke være koblet til kanten av et prosesspool, kun kanten av et black-box pool. Meldingsflyt kan indikere muligheten for kommunikasjon, og ikke sikkerheten ved det. En User task med en utgående melding kan sende meldingen, men må ikke sende den.



Pool

En pool er en rektangulær boks som representerer inneholder en enkel prosess og er en deltager i et samarbeid. To typer:

- **Prosesspool (white-box pool)** = inneholder flytelementer og merkes med navnet til prosessen (eks: Bestillingsprosess)
- **Black-box pool** = er tom og merkes med navnet til entiteten eller rollen (eks: Kunde)



Hvis diagrammet ikke viser noen meldingsflyt, trenger man ikke å tegne poolen. Ved hierarkisk modellering er det best å ikke bruke pool på barnenivå utvidelsen som tegnes i et separat diagram

Lane

Lane er en valgfri oppdeling av ett prosessnivå, som brukes for å gruppere aktiviteter etter hvem som utfører dem (*User tasks*: avdelinger eller roller). De kan også brukes av andre typer aktiviteter, slik som *Service tasks*, for eksempel kan poolen deles inn i value-adding og non-value-adding aktiviteter som utføres automatisk. Hvert sett med lanes er definert for et bestemt prosessnivå, så hvis foreldre- og barnenivå diagram skal referere til samme lanes må man tegne lanesettet i begge diagrammene.

Som vi kan se på figuren kan en lane i et prosessnivå ha et barn lanesett (dvs. lane i et lane). Dette kan brukes hvis man for eksempel har ulike roller innenfor en avdeling.



Dataobjekt og datalager

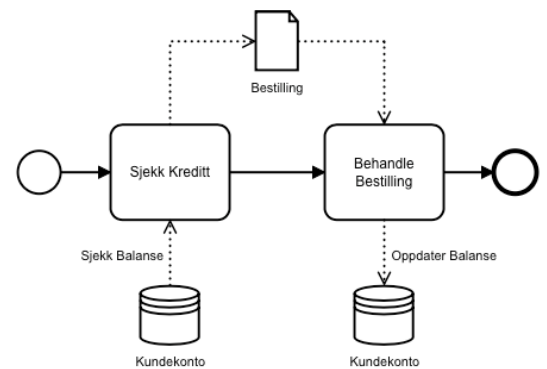
Vi skiller mellom:

- **Dataobjekt** = en lokal variabel i et prosessnivå. Det er en mengde midlertidig data som lagres i prosessinstansen, mens den kjører. Verdien til dataobjektet er synlig for andre elementer i det samme prosessnivået eller en av barnenivåene. Det kan for eksempel sendes som input til en aktivitet eller testes av en gateway betingelse. Verdien er usynlig for elementer i søsken- eller foreldrenivå. Når prosessen er slutt vil dataobjektet forsvinne. Det fungerer som en variabel i et dataprogram.
- **Datalager** = representerer vedvarende data, slik som en enhet med informasjon som lagres i en database eller business system. Det kan utsettes for queries eller oppdateres av prosessen eller entiteter utenfor prosessen. Det forsvinner ikke når prosessen slutter. Det er ofte dette prosessmodellbyggerne mener med «data».



Dataassosiasjoner kobler dataobjekter og datalager til andre elementer i prosessen. De ligner på meldingsflyter, men er prikkede linjer med V som pilhode.

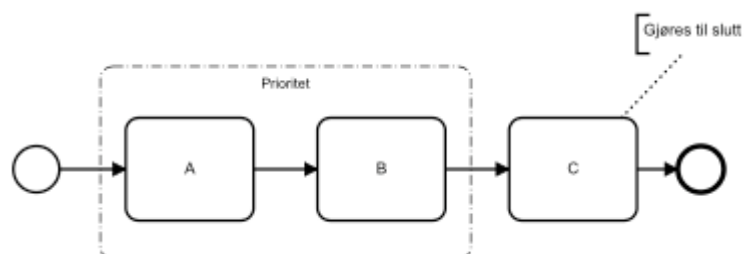
Dataobjekter kan brukes for å representere en dataflyt fra en aktivitet til en annen, ved at en data output assosiasjon går fra en aktivitet eller hendelse til dataobjektet og en data input assosiasjon går fra dataobjektet til en annen aktivitet eller hendelse (se figur). Datalager kan brukes for å representere en oppdatering av databasen ved at en dataassosiasjon går fra en aktivitet eller hendelse til datalageret. Det kan også brukes for å representere en spørring (*query*) ved at en dataassosiasjon går fra datalageret til en aktivitet eller hendelse.



Dokumentering, tekstannotering og gruppe

BPMN modellen og de fleste individuelle elementene inneholder et dokumentasjonselement i XML der man kan gi så mye informasjon man vil. Dokumentering har ingen grafisk element og vil derfor ikke vises i diagrammet. **Tekstannoteringer (merknader) kan plasseres i diagrammet for å gi tilleggsinformasjon.** De er festet til et grafisk element med en prikket linje uten pilhode. Dette er artefakter, noe som betyr at de ikke påvirker prosessflyten, men gir støttende informasjon. **Gruppe er en annen artefakt som brukes for å indikere en relasjon mellom elementer.** Det ser ut som

en stiplet boks som omfavner bestemte elementer i diagrammet. Det kan brukes som en visuell fremhever i diagrammet.



Kapittel 5 – Metoden

Metoden er ikke en del av BPMN spesifikasjonen, siden BPMN ikke har en offisiell metodikk. Den kan brukes for å lage et ikke-utførbart prosessdiagram som gir prosesslogikken på en betydningsfull måte. Metoden er et forsøk på å standardisere strukturen til en «God BPMN» modell, slik at diagrammet blir mer forståelig. **Målet til metoden er å lage en oppskrift som kan brukes for å lage en BPMN modell.** Den er basert på en **hierarkisk modelleringsstil** som viser viktige fakta om hele prosessen i et topp-nivå diagram og detaljer i barnediagram. Det bruker merkelapper for å orientere seg mellom nivåene.

Målet til metoden er å lage en «God BPMN» modell som er:

- **Korrekt** = diagrammet følger reglene som gis av BPMN spesifikasjonen
- **Tydlig** = logikken til prosessen er entydig og åpenbart forklart i diagrammet. Modellen skal ikke avhenge av dokumentasjon ved siden av. Prosesslogikken beskriver hvordan man går fra en oppgave til neste.
- **Fullstendig** = diagrammet gir flyten av aktivitet, hvordan prosessen starter, signifikante slutttilstander og kommunikasjon med eksterne entiteter
- **Konsistent** = hvis de gis samme fakta om prosesslogikken, skal det kunne lages flere modeller som er tilnærmet like.

Endetilstand er et viktig begrep i metoden. En aktivitet er en handling som utføres gjentatte ganger. **Hver instans av en aktivitet vil ha en veldefinert start- og endetilstand.** Hvis prosessen har tre mulige neste steg, kan man bruke en aktivitet med tre ulike endetilstander og deretter en gateway som sjekker endetilstanden og deler prosessflyten inn i tre ulike baner. Hvis aktiviteten er en oppgave, vil endetilstandene være usynlig i diagrammet. Hvis aktiviteten er en subprosess kan endetilstandene representeres av ulike slutthendelser som er synlige i diagrammet. Navnet på gateways bør matche en av endetilstandene.

For å illustrere metoden, skal vi se på et bilforhandler-scenario der en ny bil blir kjøpt. Prosessen beskrives fra perspektivet til bilforhandleren.

Steg 1 – Definer prosessomfanget

Top-down modelleringen begynner med at man **definerer omfanget til prosessen, altså hvor den starter og slutter.** Viktige spørsmål er:

1. **Hvordan starter prosessen?** Er det en ekstern forespørsel eller planlagt prosess?
2. **Hva bestemmer når prosessen er ferdig?** Når en instans er ferdig kan den ikke være involvert i flere handlinger.
3. **Hva representerer hver prosessinstans?** Vi ser på starthendelsen, for eksempel vil instansen være en respons på en forespørsel hvis starthendelsen mottar en forespørsel
4. **Kan prosessen ende på ulike måter?** Har prosessen flere endetilstander?

For bilforhandler-scenariet blir svaret på spørsmålene følgende:

1. Prosessen starter med en bestilling, som er en avtale som sendes fra en bestemt kjøper som ønsker å kjøpe en bestemt bil for en avtalt pris
2. Prosessen er ferdig når hele kjøpet er betalt og kjøperen mottar bilen
3. En instans ved prosessen vil være en enkelt bestilling
4. Prosessen har flere mulige måter den kan ende på: (1) Transaction complete som er den normale suksessfulle endetilstanden, (2) Financing unavailable som er endetilstanden der kunden ikke klarer å gi sikker finansiering og (3) Delivery Data

Unacceptable som er endetilstanden der kunden avbryter bestillingen som følge av at leveringsdatoen ble for sen.

Det kan være nyttig å dele opp endetilstandene fordi det kan avsløre problemer ved ulike deler av organisasjonen. For eksempel kan Financing unavailable foreslå at det er problemer ved finansavdelingen til organisasjonen, siden bestillingen kun skal initieres hvis kunden blir tolket som kredittverdig. Delivery Data Unacceptable kan foreslå at det er problemer ved salgsavdelingen siden den estimerte leveringsdatoen ble feil. **Ved å identifisere disse som distinkte mislykkede endetilstander, kan vi forstå årsaken bak hvorfor prosessen mislyktes.**

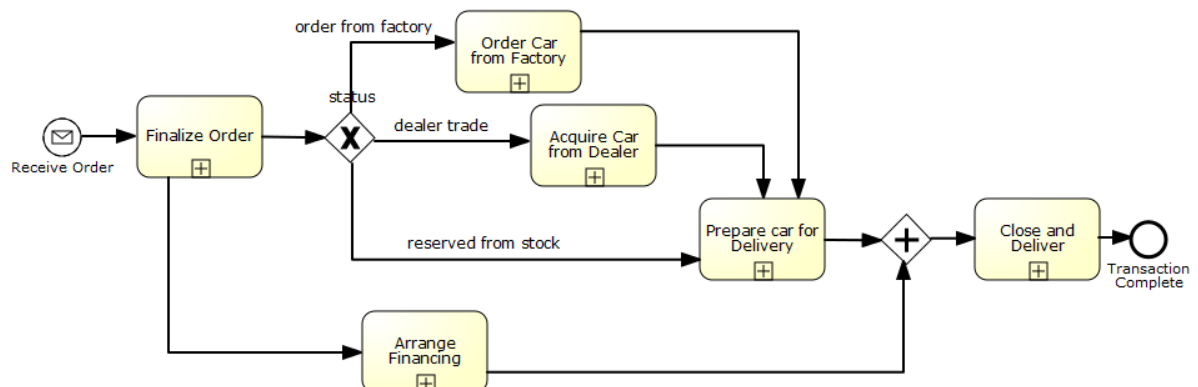
Steg 2 – Lag topp-nivå diagrammet for *happy path*

Neste steg i metoden er å lage topp-nivå diagrammet for happy path. Dette er en nummering av hovedaktivitetene til prosessen (10 eller færre) for å representere topp-nivå BPMN diagrammet. Disse hovedaktivitetene plasseres i en liste som deretter brukes for å lage et topp-nivå diagram. Hovedaktivitetene kan ses på som beholdere som skal fylles med detaljer senere.

For bilforhandler-scenariet vil hovedaktiviteten være følgende:

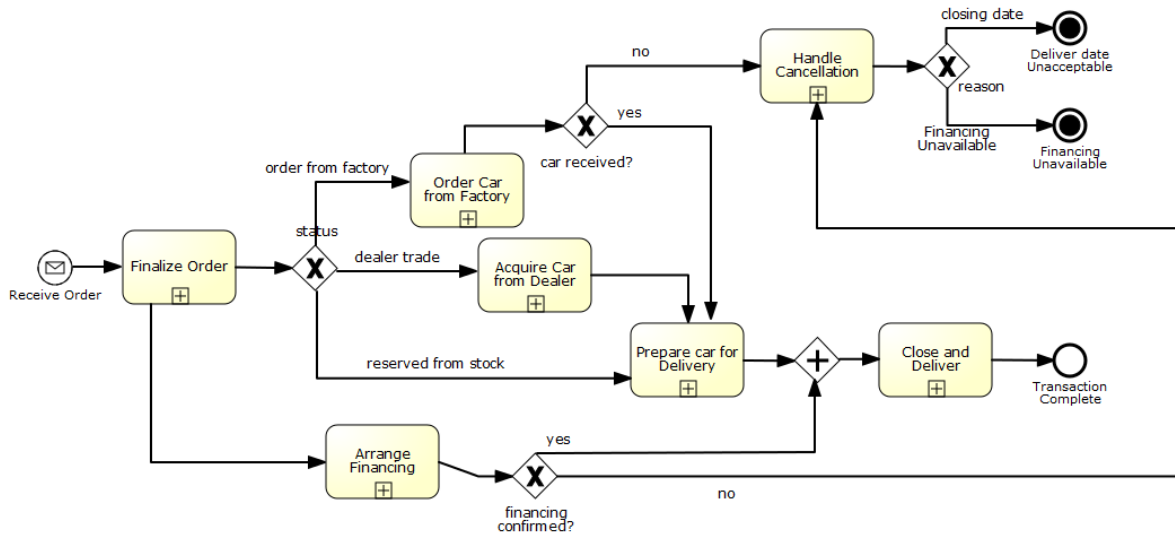
- Fullføre bestilling
 - Utføres av: salgsavdeling
 - Endetilstander: reservert fra lager, forhandlerhandel, bestill fra fabrikk
- Skaffe bil fra lokal forhandler
 - Utføres av: salgsavdeling
 - Endetilstander: mottatt bil
- Bestille bil fra fabrikk
 - Utføres av: salgsavdeling
 - Endetilstander: mottatt bil, bestilling kansellert
- Forbered levering av bil
 - Utføres av: service avdeling
 - Endetilstander: klar
- Ordne finansiering
 - Utføres av: finansavdeling
 - Endetilstander: finans bekreftet, finans utilgjengelig
- Lukk og lever
 - Utføres av: salgsavdeling
 - Endetilstander: bil mottatt

Dette gir følgende top-nivå happy path diagram:

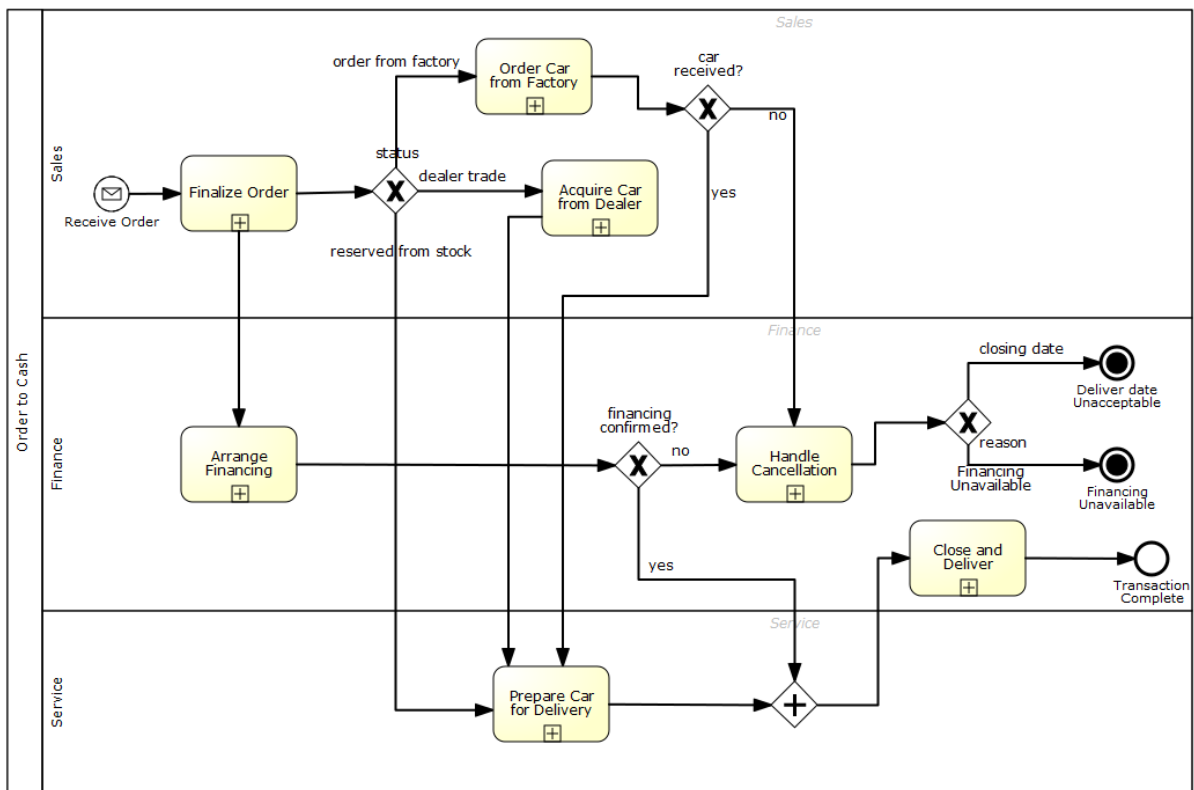


Steg 3 – Legg til topp-nivå unntaksbaner

Neste steg i metoden er å legge til unntaksbaner i topp-nivå diagrammet. Alle endetilstandene til prosessen blir vist som separate slutthendelser i diagrammet. For bilforhandler-scenariet blir diagrammet:



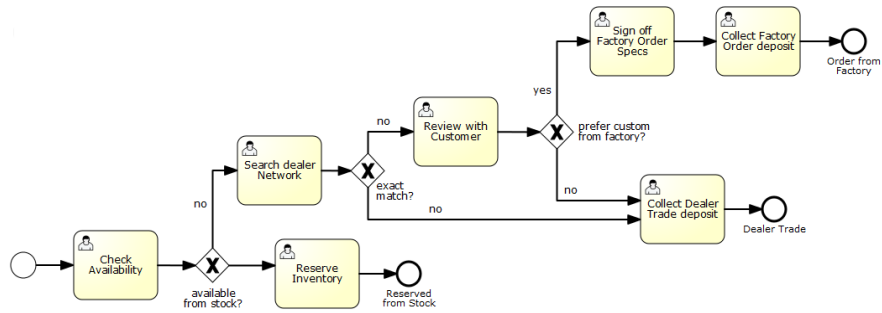
Deretter bruker vi utfører-delen av hovedaktivitetene for å lage pool og lanes:



Steg 4 – Utvid subprosesser for å vise detaljer på barnenivå

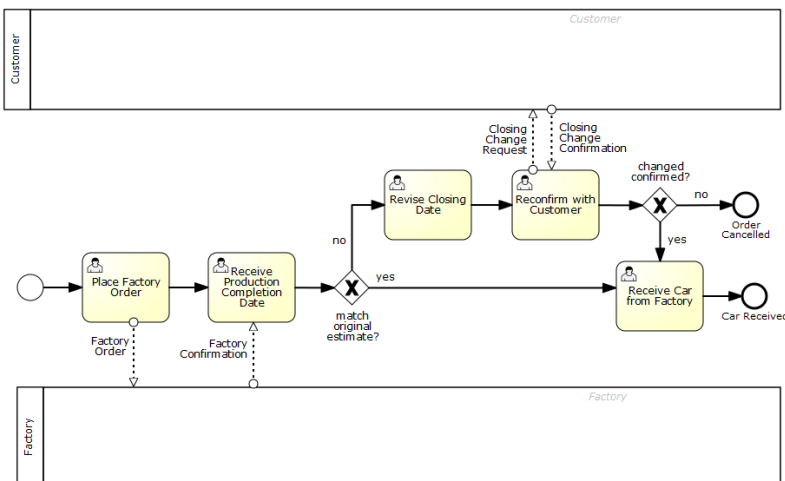
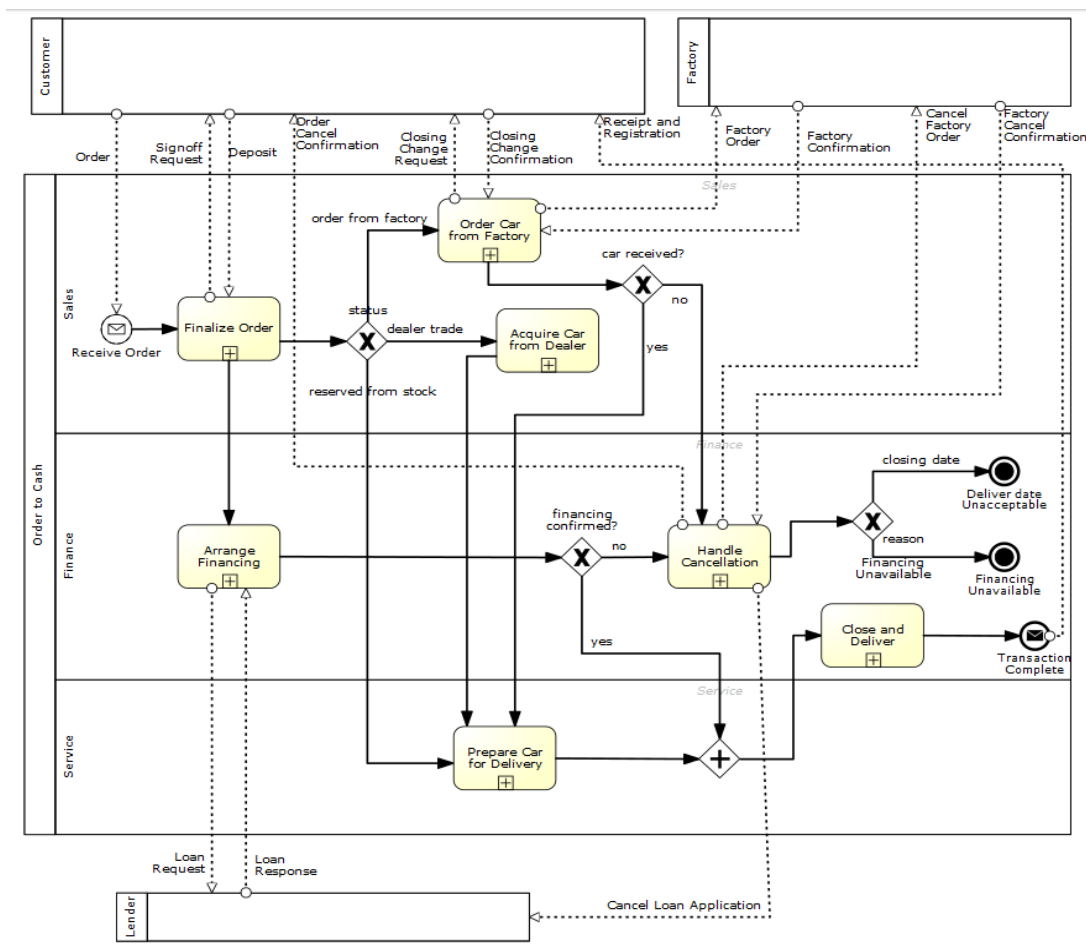
Topp-nivå diagrammet gir hvordan prosessen starter og slutter, men den gir lite informasjon om interne detaljer. Neste steg i metoden er derfor å lage barnenivå utvidelser av hver hovedaktivitet. Disse tegnes som separate diagram. Barnenivå utvidelser må ha None starthendelse og de er som regel ikke tegnet i en pool. Husk at lanes blir definert ved ett nivå og vil være uavhengig av andre nivå.

For bilforhandler-scenariet vil barnenivå diagrammet for *Finalize Order* bli:



Steg 5 – Legg til evt. meldingsflyter til eksterne pools

Legg til eventuelle meldingsflyter som kobler prosessen til eksterne black-box pools. En BPMN modell må ikke nødvendigvis ha meldingsflyt, men dette gir verdifull informasjon om business konteksten til prosessen, for eksempel hvordan prosessen kommuniserer med kunder, leverandører, osv. Meldingsflyten ved foreldre- og barnenivå diagram bør være lik.



Her kan vi se barnenivå diagrammet for *Order car from factory*. Legg merke til at dette diagrammet har samme meldingsflyt som topp-nivå diagrammet, ved at antall piler og navn på disse er like.

Kapittel 6 – BPMN stil

Metoden brukes for å sikre at BPMN modellen er konsistent , men alene er den ikke tilstrekkelig for å sikre at modellen har tydelig prosess logikk og er uavhengig av dokumentasjon ved siden av. For å oppnå dette trenger i BPMN stilen. Denne stilen består av et sett med stilregler som kan brukes for å sjekke gyldigheten til modeller.

Grunnleggende prinsipp ved BPMN stilen

Det grunnleggende prinsippet ved BPMN stilen er at prosesslogikken bør være entydig fra diagrammet alene. Det er dette som menes med «God BPMN». Prosesslogikken er logikken til sekvensflytene, for eksempel når en aktivitet ender hva skjer, under hvilke betingelser?

BPMN stil handler om å bruke de visuelle elementene i diagrammet på beste mulige måte.

Merking er en av de viktigste delene av BPMN stilen.

Stilregler

Følgende er stilreglene:

1. Bruk ikoner og navn for å gjøre prosesslogikken mer tydelig
2. Bruk hierarkisk modellering, slik at hvert prosessnivå får plass på ett ark
3. Bruk black-box pool for å representere eksterne deltagere, slik som kunder eller andre eksterne forespørre og tjenesteleverandører
4. Begynn kundevedtatte prosesser med en message starthendelse som mottar en meldingsflyt fra kundens black-box pool
5. Modeller interne prosessdeltagere (aktive utførere) som lanes i ett prosesspool, og ikke som separate pools. Merk lanes med rollen eller organisasjonsenheten som utfører aktivitetene. Separate pools indikerer uavhengige prosesser (brukes når det ikke er 1:1 relasjon mellom prosessinstansene)
6. Merk prosesspool med prosessnavnet og black-box pool med deltagerens rolle eller business entiteten
7. Lag separate sluttendelser for suksessfulle og mislykkede endetilstander ved prosessen, og gi dem navn som indikerer endetilstanden.
8. Merk aktiviteter som VERB-Substantiv
9. Bruk starthendelse trigger for å gi hvordan prosessen starter. Message starthendelse merkes med Motta X, timer starthendelse merkes med prosessplanen og None starthendelse (manuell start) merkes med ingenting
10. Hvis subprosessen følges av en gateway med et ja/nei spørsmål, skal subprosessen ha flere sluttendelser og en av dem skal matche gateway merkingen.
11. Hvis meldingsflyt for alle message hendelser
12. Match meldingsflyten (antall og navn) i foreldre- og barnenivådiagram
13. Merk meldingsflyten med navnet til meldingen (ikke bruk dataobjekt mellom pools)
14. Bruk ulike navn på sluttendelser i samme prosessnivå. Like endetilstander slås sammen til en sluttendelse
15. Bruk ulike navn på aktiviteter og ikke la subprosess inneholde aktivitet med samme navn som subprosessen. Hvis det er samme aktivitet, bruk *Call Activity*
16. Bruk kun én None starthendelse i subprosesser (unntak er parallell boks). Alternative starthendelser krever triggerikon og kan derfor kun brukes i topp-nivå prosess
17. Prosessnivå pool i barnenivå diagram skal merkes med navnet til topp-nivå prosessen og ikke navnet til subprosessen (bør utelate pool i barnenivå diagram)
18. Barnenivå diagram skal ikke inneholde aktiviteter fra noen andre prosesser enn foreldrenivå prosessen. Toppnivå diagram kan inneholde elementer fra flere prosesser

19. Ikke bruk XOR-join for å slå sammen alternative baner, hvis ikke det er inn i en annen gateway. Koble sekvensflytene direkte til aktiviteten eller hendelsen
20. Ikke bruk AND-join for å slå sammen parallelle baner i None slutthendelse, fordi en slik slutthendelse impliserer alltid join
21. Følg BPMN regler:
 - a. En sekvensflyt kan ikke krysse en pool eller en subprosess
 - b. En meldingsflyt kan ikke koble noder i samme prosess
 - c. En sekvensflyt kan bare koble sammen aktiviteter, gateways og hendelser, og begge ender må være tilkoblet
 - d. En meldingsflyt kan bare koble sammen aktiviteter, message/multiple hendelse eller black-box pool, og begge ender må være tilkoblet.
 - e. Alle aktiviteter, gateways og hendelser må være koblet til en kontinuerlig kjede av sekvensflyter fra starthendelsen til en slutthendelse

Evaluering

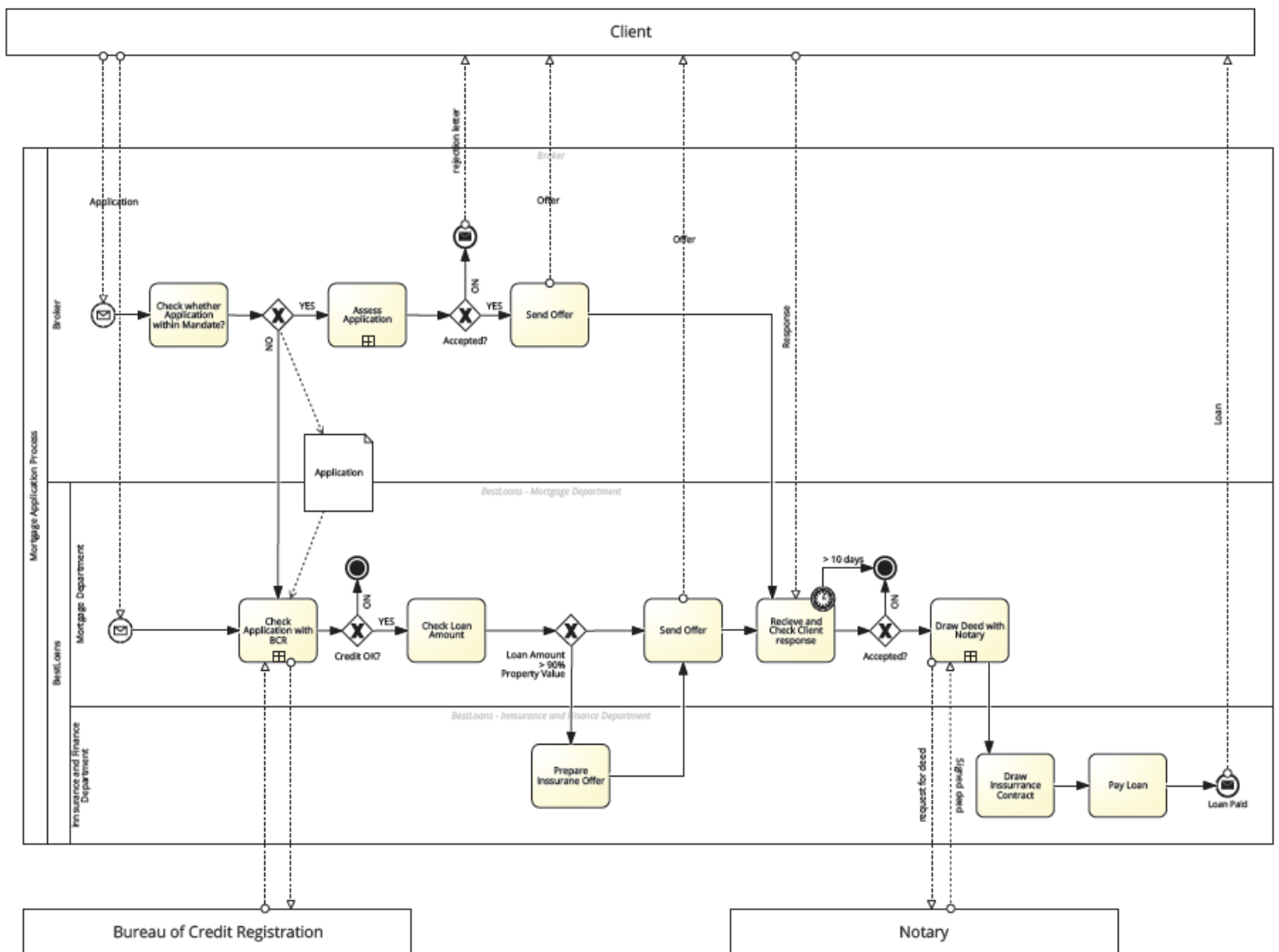
Når man evaluerer en BPMN modell ser man på:

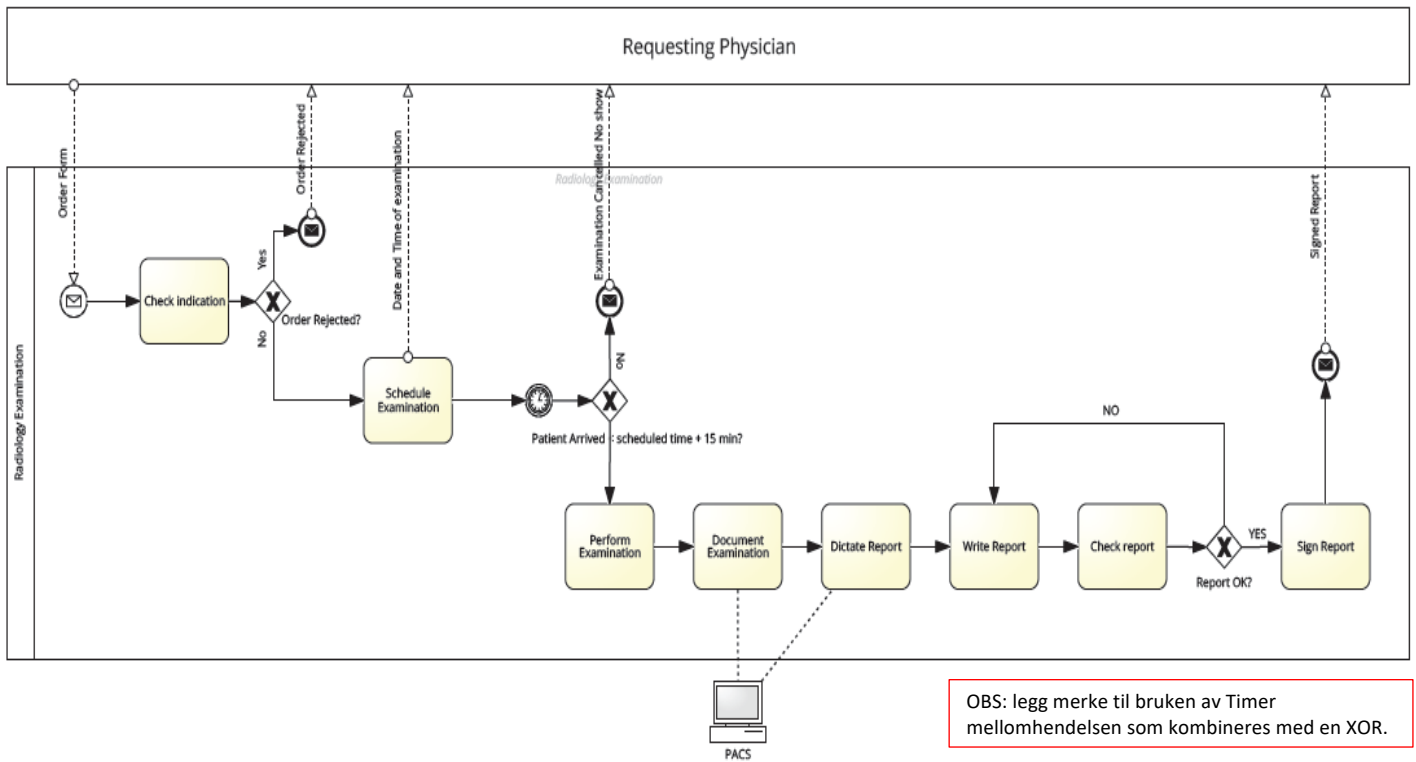
- **Empirisk kvalitet** = layout, ingen kryssende linjer, stil (venstre til høyre, splitt og join, navngivning, osv.)
- **Syntaks** = blir BPMN brukt riktig?
- **Semantisk** = er hele casen modellert? Er det noe som motsier beskrivelsen av casen?

Løsning på individuelle øvinger

Følgende er løsningsforslaget til de individuelle øvingene.

OBS: legg merke til bruken av timeren på aktiviteten som mottar og sjekker klientresponsen. Dette representerer en XOR der sekvensflyten med timeren utføres hvis det ikke blir mottatt noen respons innen tidsfristen. Hvis det mottas en respons blir kun den andre sekvensflyten aktivert. Dvs. kun én av de utgående sekvensflytene blir aktivert.

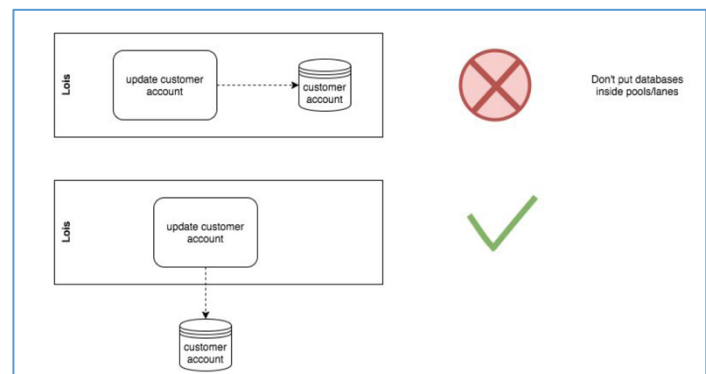
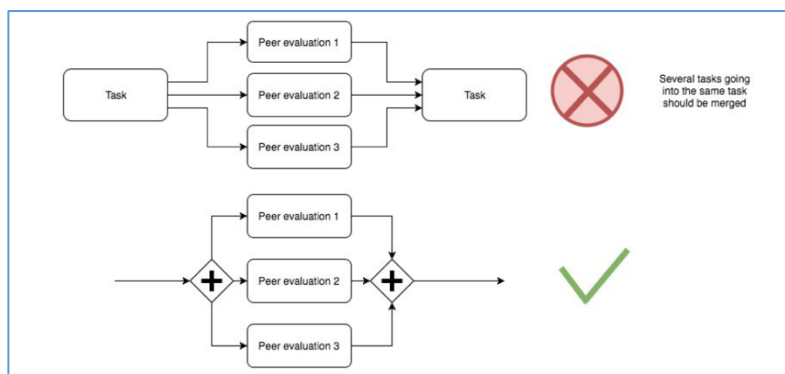
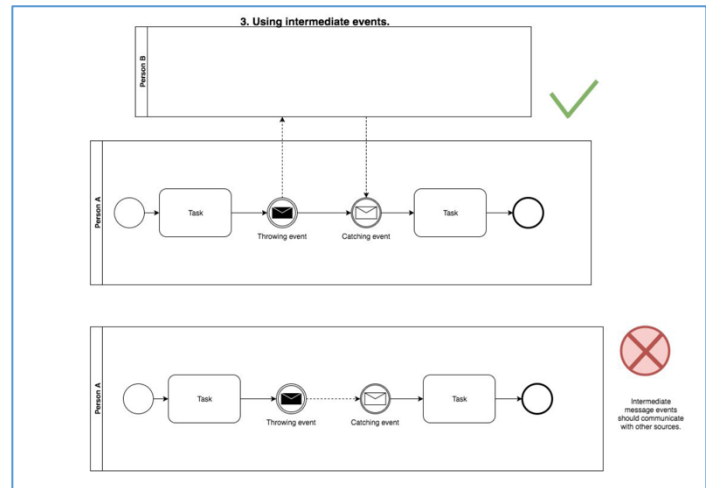
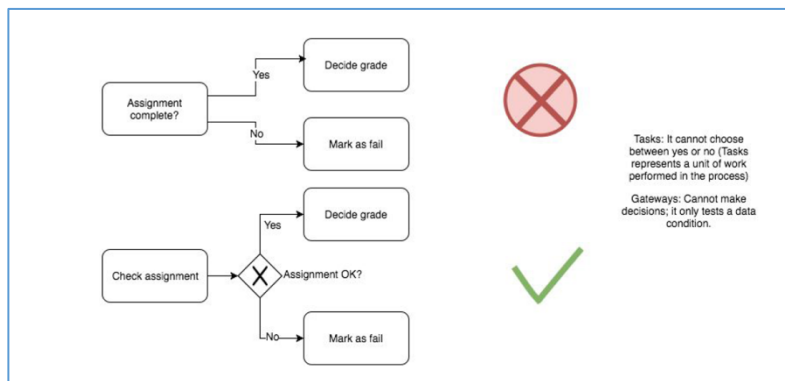




Vanlige feil

Følgende er vanlige feil som blir gjort ved BPMN modellering:

- Aktiviteter blir ikke merket som Verb-Substantiv eller verb er ikke imperativ form
- Slutthendelse er ikke merket med informativt navn
- Layout er venstre til høyre eller nedover
- Mange linjer som krysser hverandre
- Manglende gateways joins når man har splitt
- Vanskelig å forstå navn
- Mange starthendelser for ulike underdeler
- Manglende AND-gateway
- Manglende XOR-splitt før aktivitet



IS metodikk og Enterprise arkitektur

Denne delen av kompendiet er basert på kapittel 2.1 og 2.2 i boka Model-based Development and Evolution of Information Systems: A Quality Approach av John Krogstie.

I dette dokumentet blir IS metodikk brukt for å betegne prosjektet der man bruker ulike metoder og teknikker for å lage et informasjonssystem. Dette prosjektet involverer analyse, kravspesifisering, design og implementering av informasjonssystemet. Noen IS metodikker er Waterfall modellen, iterativ prototyping, spiral modellen, smidige metoder, osv.

2.1 Et rammeverk for IS metodikker

Det har blitt utviklet flere rammeverk for klassifisering av IS metodikker, men ulempen ved disse er at de har begrenset omfang, siden de i hovedsak ser på utviklingen av ett enkelt applikasjonssystem i et relativt stabilt miljø. Mange organisasjoner er avhengig av en rekke applikasjonssystem som støtter nåværende og fremtidige oppgaver, og nykommere er avhengig av å raskt etablere slike system. Det blir også en stadig større variasjon i organisasjonsformer, for eksempel virtuelle virksomheter, value chain nettverk, osv. **Organisasjoner er under et konstant press for endring fra både interne og eksterne krefter.** Applikasjonssystemene som brukes av organisasjonen må også utsettes for raske endringer for at organisasjonen skal kunne beholde konkurranseevnen. **Endringer er altså det normale, og ikke unntaket for en organisasjon.** Når man skal klassifisere IS metodikker, er det derfor flere elementer som må vurderes.

Med utgangspunkt i dette, kan IS metodikker klassifiseres basert på følgende områder:

- **Målet ved metodikken** = hva er målet ved metodikken og hvilken verdi forventer man å oppnå?
- **Prosessdekning** = når blir metodikken brukt? Hva er hovedoppgavene som metodikken dekker?
- **Produktdekning** = hvilken del av applikasjonsporteføljen til informasjonssystemet dekkes av metodikken?
- **Evnebehov** = hvilke evner trengs for å få metodikken til å fungere?
- **Involverte deltagere** = hvem er involvert?
- **Organisering** = hvordan er arbeidet organisert?
- **Lokasjon av endring** = hvor tar endringene sted?

Målet ved metodikken

For å klassifisere en IS metodikk kan man se på hvilket mål metodikken har. En IS metodikk kan brukes for å:

- **Sikre økonomisk gevinst**
- **Sikre personlig gevinst**
- **Sikre organisasjon (business) gevinst**
- **Sikre samfunnsgevinst**

Disse målene handler altså om å oppnå en form for verdi. For eksempel kan man sikre en organisasjonsgevinst ved å øke verdiene som bestemmer helsen til organisasjonen på lang sikt (eks: ansattverdi). For å oppnå verdi vha en IS metodikk kan følgende steg utføres:

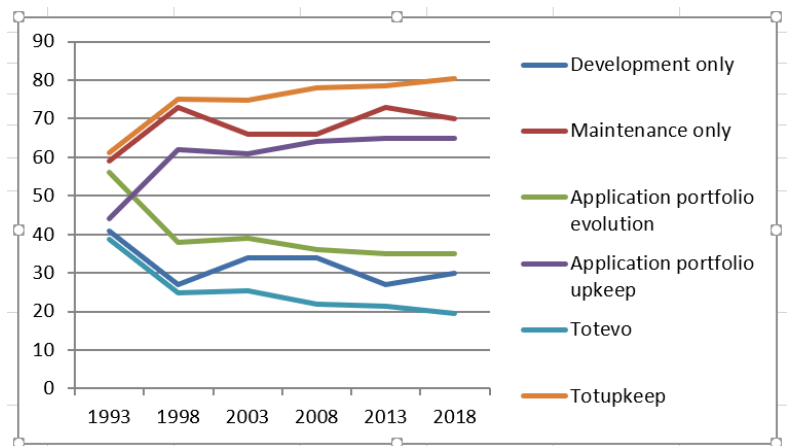
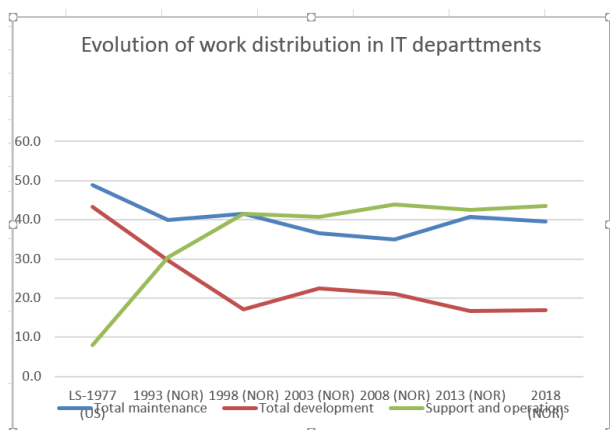
1. **Planlegge for verdioppnåelse** (utvikle business case og realiseringsplan)
2. **Utføre endring**
3. **Implementer endring og realiser verdien**

Prosessdekning

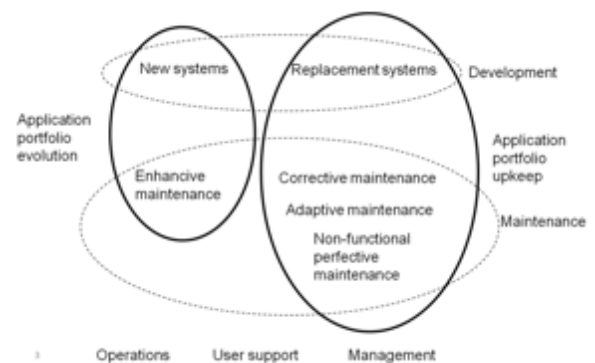
For å klassifisere en IS metodikk kan man se på når metodikken blir brukt og hvilke hovedoppgaver den skal dekke. En IS metodikk kan brukes for å dekke en av følgende hovedtype av oppgaver:

- **Utvikling av applikasjonssystem**
- **Bruk av applikasjonssystem i produksjon**
- **Drift av applikasjonssystem i produksjon**
- **Vedlikehold og evolusjon av applikasjonssystem i produksjon**

Tradisjonell programvareutvikling og IS metodikker har et kortsiktig perspektiv og dekker i hovedsak den første typen, altså utvikling av applikasjonssystem. Enterprise arkitektur har et lengre perspektiv. En omfattende metodikk bør dekke alle fire områdene, utvikling, bruk, drift og vedlikehold, på en integrert måte.



Grafene viser at det meste av arbeidet som gjøres innenfor en IT avdeling er forbundet med vedlikehold. **S skillet mellom utvikling og vedlikehold er ikke nødvendigvis så stort.** Som vi har sett er det normalt med endringer i organisasjonen, noe som også involverer endringer i IS. Halvparten av arbeidet som gjøres i sammenheng med vedlikehold er derfor videre utvikling av informasjonssystemets portefølje. I tillegg er nesten halvparten av de nye systemene som blir utviklet erstatningssystem, som opprettholder porteføljen til applikasjonen istedenfor å legge til ny funksjonalitet. Derfor kan det være bedre å se på hvor mye metodikken er fokusert på å utvikle porteføljen, istedenfor å skille mellom vedlikehold og utvikling. Det har blitt observert at moderne utviklingstyper har ført til mindre vedlikehold, noe som indikerer at det kan være nyttig å bruke en metodikk som forener utvikling og vedlikehold.



Produktdekning

For å klassifisere en IS metodikk kan man se hvilken del av applikasjonsporteføljen metodikken dekker, altså hvilket produkt IS metodikken utvikler, bruker, drifter og/eller vedlikeholder. Dette produktet kan være:

- Et enkelt applikasjonssystem
- En familie av applikasjonssystem
- Hele porteføljen av applikasjonssystem i organisasjonen
- Hele organisasjonen (utvikling og vedlikehold av enterprise arkitekturer)

- En klynge samarbeidende og delvis konkurrerende organisasjoner i kombinerte digitale økosystem

For endrebrukeren av informasjonssystemet er det ikke viktig hvilket applikasjonssystem som har blitt endret, men heller at dens behov er dekket av den fullstendige porteføljen av applikasjonssystem. **Applikasjonssystem blir ikke utviklet i et vakuum, siden de har relasjoner til gamle system og blir integrert inn i andre system.** Mer enn halvparten av nye system er erstatningssystem som derfor vil være basert på de gamle systemene.

Applikasjoner i organisasjoner blir mer og mer integrert ved at de bruker felles ressurser og komponenter, og 73% av alle applikasjoner er avhengig av data som produseres av andre system. Organisasjoner kan ha mange hundre applikasjoner og en av de viktigste årsakene for erstatning av system er et ønske om integrering med andre system. Selv om metodikken brukes for å utvikle ett enkelt eller en familie av applikasjonssystem, er det derfor viktig at den uansett ser på hele porteføljen av applikasjonssystem i organisasjonen.

Evnebehov

For å klassifisere en IS metodikk kan man se på hvilke evner som trengs for at metodikken skal fungere. Oppgavene innenfor en metodikk krever bestemte egenskaper ved både personlig- og organisasjonsnivå, for å være sikker på at målet til metodikken oppnås. For eksempel for Scrum metodikken trengs det en produkteier som kan prioritere egenskaper, akseptere og avslå resultater, osv., en Scrum Master som kan sikre samarbeid, skjerme teamet fra forstyrrelser, osv. og et team som kan demonstrere resultatet, organisere seg selv, osv. For RUP metodikken trengs det evner innenfor analyse, kravspesifisering, design, testing, osv.

For enterprise arkitektur og enterprise modellering, er følgende egenskaper nødvendig:

- Evnen til å modellere
- Evnen til å velge passende EM tilnærming og skreddersy den til situasjonen
- Evnen til å intervju involverte domeneeksperter
- Evnen til å definere et relevant problem
- Evnen til å definere krav ved resultatet
- Osv.

Involverte deltagere

For å klassifisere en IS metodikk kan man se på hvilke deltagere som er involvert i metodikken. I sammenheng med utvikling og evolusjon av informasjonssystem er det et stort antall mennesker som har ulike roller. Interessenter i implementering av informasjonssystem kan deles inn i følgende grupper:

- De som er ansvarlig for design, utvikling, introduksjon og vedlikehold, for eksempel prosjekt manager, systemutviklere, osv.
- De som har finansinteresse og er ansvarlig for salg eller kjøp av applikasjonssystem. De kalles kunder av systemet
- De som bruker systemet, for eksempel direkte brukere (endebrukere) og indirekte brukere
- De som påvirkes av langsiktige effekter ved systemet

For metodikker vil det variere hvordan de ulike deltagerne deltar i aktiviteter som involverer produksjon og utvikling av informasjonssystemet. Ved **direkte deltagelse** har alle interessenter en mulighet til å delta, mens ved **indirekte deltagelse** vil hver deltager delta via representanter som passer på deres interesse. Representantene er enten valgt av noen

(eks: management) eller stemt frem. Deltagelse kan gi økt eierskap og motivasjon blant interessenter, mer transparent prosess, bedre karakterbygging, forbedret løsning, osv. Verdien ved økt deltagelse kan variere.

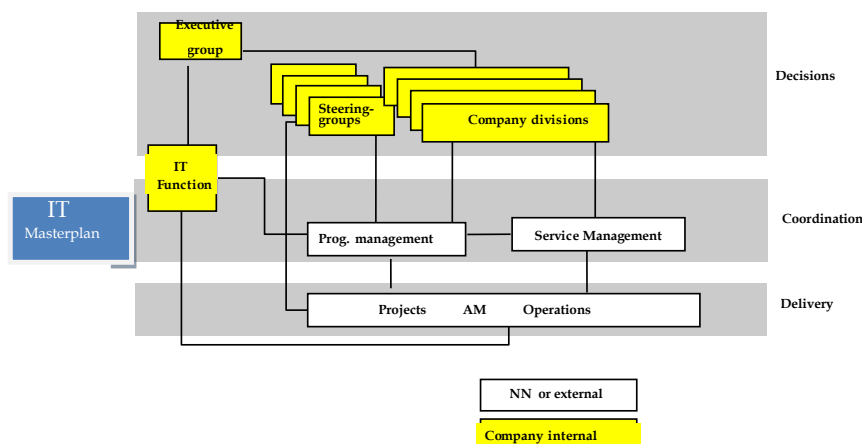
Påvirkningen til deltagere kan deles inn i følgende nivåer:

- -1 = interessenter blir gitt feil eller skjev informasjon
- 0 = interessenter blir gitt null informasjon
- 1 = interessenter blir gitt informasjon
- 2 = det holdes konsultasjoner med interessenter
- 3 = det tas hensyn til råd fra interessenter
- 4 = interessenter får delta i avgjørelser sammen med utviklere
- 5 = interessenter får autoritet
- 6 = problemeiere får full kontroll

Jo høyere på skalaen, desto mer ansvar får interessentene.

Organisering

For å klassifisere en IS metodikk kan man se på hvordan arbeidet er organisert i metodikken. Organisasjonene som er involvert i utviklingen og evolusjonen av informasjonssystemet kan være ordnet på ulike måter. I begynnelsen var IT flettet sammen med andre aktiviteter i organisasjonen, men etterhvert som IT ble viktigere ble det etablert en egen IT-avdeling som senere ble delt opp i flere avdelinger (eks: utvikling, vedlikehold, drift, brukerstøtte og management. Deler av disse aktivitetene har blitt outsourcet til distinkte enheter i organisasjonen eller eksterne partnere. Det har blitt vanligere med open source, co-utviklet programvare som er tilgjengelig gratis (eks: OSS), digitale økosystem, osv. Figuren under viser et eksempel på hvordan en metodikk kan fordele oppgaver blant ulike deler av organisasjonen (gul) eller eksterne partnere (hvit, outsourcing)



Lokasjon av endring

For å klassifisere en IS metodikk kan man se på hvor endringene tar sted. IS metodikker kan ha ulike fremgangsmåter på hvor arbeidet forbundet med endringene foregår. I begynnelsen var det vanligst at systemer ble utviklet internt (*inhouse* = vha. egne ansatte) eller ved å bruke midlertidige konsulenter. I de siste 10 årene har det blitt vanligere med outsourcing, og 91% av organisasjoner outsourcet deler av IT aktiviteten i 2013. Det er likevel veldig få som outsourcer all IT aktivitet. Tabellen viser hvor stor andel av ulike IT aktiviteter som ble outsourcet av organisasjoner i 2013. De fleste informasjonssystemene blir utviklet «en annet sted». Det har blitt utviklet nye leveringsmetoder, slik som infrastruktur på skyen, OSS komponenter, programvare økosystem, osv.

Activity	Private	Public	Total
IT in total	37	30	33
Development	32	57	48
Maintenance	33	36	34
Operations	44	28	33
User-support	13	18	16

2.2 En kort historie om IS metodikker

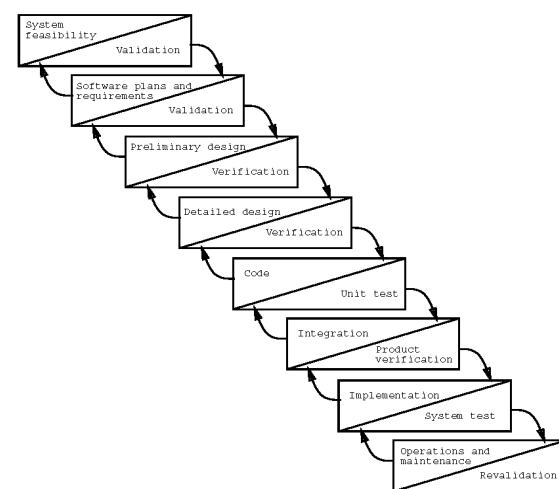
De fleste metodikkene inneholder følgende abstraksjonsnivåer:

- **Analyse**
- **Kravspesifikasjon**
- **Design**
- **Implementering av system**

I de tidligere årene med databehandling (1945-1960) brukte alle en naiv tilnærming kalt *code and fix*, der det var fokus på programmering, kompilatorer, opererende system, osv. På 60-tallet begynte Langefors å forstå behovet for abstraksjonsnivåer, og han delte systemutviklingen inn i problemorientert analyse, produktorientert design og implementasjon. I 1968 kom begrepet *Software Engineering* som følge av en NATO konferanse og i 1970 ble Waterfall modellen brukt for første gang i IT-prosjekter. Det var den første prosjektmodellen for IT utviklingsprosjekter.

Waterfall modellen

Waterfall modellen ble først tatt i bruk i 1970, og den organiserer prosjektet som en lineær sekvens av faser, der hver fase blir fullstendig fullført ved å dokumentere dens prestasjoner. Neste fase starter når den forrige er fullført. Feedback loops mellom etterfølgende faser gjør at man kan endre dokumentasjonen fra forrige fase. Fordelene ved waterfall metoden er at den får utviklerne til å spesifisere systemet før de lager det, det fremmer design av systemkomponenter før de kodes, progresjonen er enklere å kontrollere og skriving av dokumenter gjør testing og dokumentasjon enklere (reduserer vedlikeholdskostnader).



I 1982-83 ble Waterfall modellen kritisert:

- Fasene er kunstige konstruksjoner, og en spesifikk type prosjektledelsesstrategi som er pålagt programvareutvikling. I praksis er det vanskelig å separere spesifikasjoner fra design og implementering
- Det er først ved enden av prosjektet at systemet kan kjøre. Det kan ta lengre tid før feil blir oppdaget, slik at det blir mer kostbart å fikse. Kunden og endebbrukere kan bli utålmodige og miste interesse for prosjektet. Det er vanskelig å endre systemet underveis, så det gir dårlig støtte for system evolusjon.

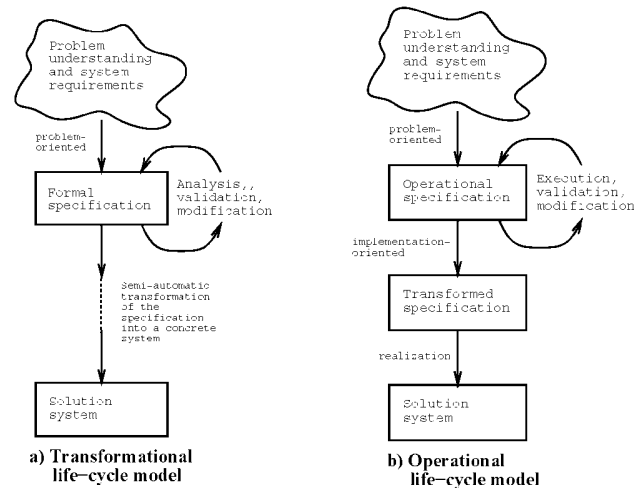
Prototyping

Prototyping som en metodikk er en svært iterativ prosess med mye bruk av prototyper. Det ble introdusert i 1990, og målet er å klargjøre bestemte egenskaper ved applikasjonssystemet ved å lage en artefakt som kan kjøre. Prototypen blir oppdatert basert på feedback fra brukeren og annen kunnskap. Prosessen blir gjentatt helt til prototypen aksepteres av brukeren. Det er altså stort fokus på brukerinvolvering og representasjon av utvalgte brukerkrav ved tidlige punkt i prosessen. Den iterative genereringen og valideringen av utførbare modeller gjør at tilnærmingen er nyttig når kravene er ustabile (endrer seg i løpet av prosessen) eller usikre. Noen fordeler ved prototyping er at det kan redusere usikkerhet, er raskere og gir bedre kommunikasjon med bruker. Noen ulemper er at det gir lite oversikt og det kan hende prototypen ikke viser hvordan systemet vil fungere i realiteten.

Transformasjons- og operasjonsutvikling

Transformasjonsutvikling ble introdusert i 1991, og det bruker verktøy for automatisk transformasjon som tar inn formelle kravspesifikasjoner og omdanner det til kode. En rekke transformasjoner blir brukt for å nå den endelige koden og i løpet av disse må en utvikler legge til flere detaljer spesifikasjonen.

Transformasjonssekvensen lagres i et register som senere kan brukes for vedlikehold eller gjentatt implementering av de formelle spesifikasjonene. Ulempen er at det krever formelle spesifikasjoner som kan være vanskelig å lage.



Operasjonsutvikling ble introdusert i 1984, og det bruker operasjonelt spesifisert språk for å lage utførbare systemmodeller (prototyping). Systemmodeller som kan kjøre blir tilgjengelig tidlig i prosessen, men de fungerer ikke som den endelige programkoden. Når modellen er ferdig vil en rekke transformasjoner utføres for å lage en annen spesifikasjon som kan tolkes av en prosessor, slik at systemløsningen blir produsert. En fordel med disse metodene er at den endelige koden er gyldig og trenger ikke verifisering, siden transformasjonene bevarer korrektheten.

Spiral modellen

Spiral modellen ble introdusert i 1988, og det er basert på flere iterasjoner gjennom flere grunnleggende steg. Hver iterasjon omfatter noen mål som skal oppnås og består av:

1. Bestem målet for iterasjonen
2. Identifiser og evaluer alternative baner for å nå målet
3. Identifiser og evaluer risikoer for den valgte banen. Utvikle produktet
4. Evaluer resultatet og planlegg neste iterasjon.

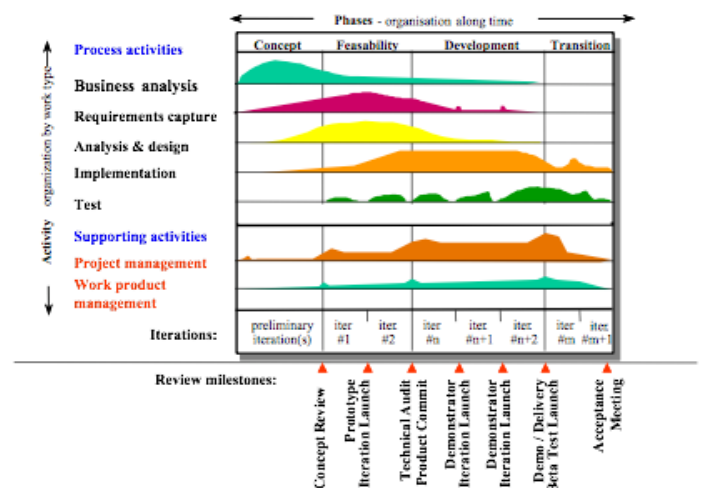
Styrken til spiral modellen er at den fokuserer på gjenbruk av programvare, den legger til rette for endringer og evolusjon, den er fokusert på å tidlig eliminere feil, osv. Ulempen er den er vanskelig å bruke for uerfarne utviklere og det er ingen tilgjengelige teknikker for ordentlig risikoanalyse.

Objektorientert systemutvikling

Objektorientert programmering ble mer populært på 90-tallet, og dermed ble også objektorienterte utviklingsmetoder mer populære (eks: RUP). Figuren viser hovedaktivitetene som utføres i løpet av en RUP systemutviklingsprosess. Det er fire faser i utviklingen: *inception*, *elaboration*, *construction* og *transition phase*. Når en fase er fullført betyr det at produktet har nådd en viss grad av fullstendighet og har nådd en milepæl (se bunnen av figuren).

Aktivitetene deles inn i prosessaktiviteter som er direkte koblet til systemutviklingen og støttende aktiviteter som støtter prosessaktivitetene. Som vi kan se på figuren vil det variere når i systemutviklingen de ulike aktivitetene er viktige. Som regel vil analyse og design dominere starten, mens implementasjon og testing dominerer slutten.

Figuren viser et eksempel, så for hvert prosjekt vil det variere hvordan kurvene ser ut.



Inkrementell og smidig utvikling

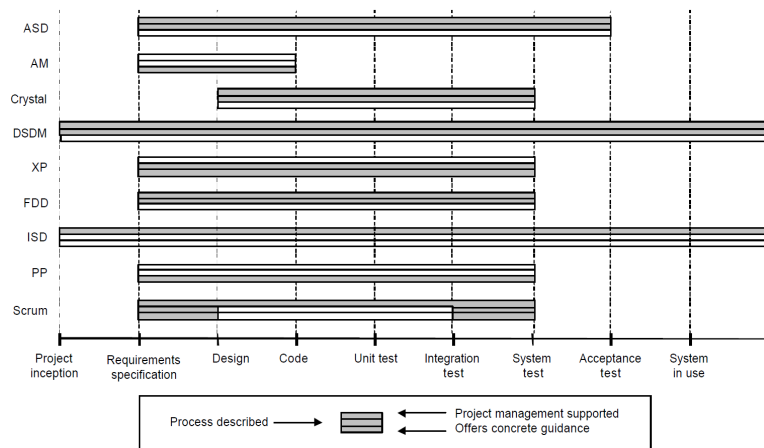
Inkrementell utvikling ble introdusert i 1988) og det er prosessen av å lage en delvis implementasjon av et totalt system og sakte legge til økt funksjonalitet eller ytelse.

Inkrementene blir gitt ut en etter en, så derfor kan det kalles stegvis utvikling. Noen fordeler er at det legger til rette for rask feedback fra brukeren og det kan lettere tilpasses endringer (system evolusjon). **Moderne tilnærminger til inkrementell utvikling kalles smidige metodikker.** Smidig utvikling er basert på at systemutviklingen skal være en sosial aktivitet med mye kommunikasjon mellom flere utviklere og brukere av systemet. Kvaliteten økes ved å bruke aktiviteter slik som parprogrammering, testdrevet utvikling, osv. Den smidige utviklingen er basert på **Agile manifesto** (2001), som definerte følgende smidige verdier:

- **Individer og interaksjoner foran prosesser og verktøy**
- **Fungerende programvare foran omfattende dokumentasjon**
- **Kundesamarbeid foran kontraktforhandling**
- **Respondere til endringer foran å følge en plan**

Det har blitt utviklet en rekke smidige metoder slik som ekstreme programmering (XP), Scrum, Agile Modelling (AM), osv. Figuren viser at **de smidige metodene har ulik fokus på de forskjellige aktivitetene.** For eksempel kan vi se at AM er mer opptatt av kravspesifisering, design og koding, mens Scrum også involverer diverse tester.

Ulempen med inkrementell utvikling er mangelen på planlegging av hvert inkrement. Hvis det første inkrementet er dårlig, kan det resultere i at brukerne gir opp prosjektet. Resultatet av inkrementell utvikling kan også være vanskeligere å vedlikeholde og integrere (ligner *code and fix* teknikken).



Bimodal IT

I 2014 ble Bimodal IT introdusert av Gartner, som er et ledende IT forsknings- og rådgivningsfirma. De forstod at tradisjonelle utviklingsmetoder ikke er tilstrekkelig for organisasjoner som har et voksende behov for enterprise applikasjoner. **Bimodal IT strategien bruker to separate, parallelle moduser for levering av IT-system, der en er fokusert på stabilitet og den andre er fokusert på smidighet:**

- **Modus 1 = IT-sentrert**, tradisjonell og sekvensiell, med fokus på sikkerhet og nøyaktighet.
- **Modus 2 = forretnings-sentrert**, utforskende og ikke-lineær, med fokus på smidighet og hastighet

Tabellen viser verdier ved de ulike modusene. Målet med å definere og separere disse to modusene var å gjenkjenne ulike management tilnærminger, verktøy og prosesser som passer ved de to modusene. Begge modusene er essensielle for IT enterprise, fordi sammen vil de sikre substansiell verdi samtidig som de fremmer innovasjon og organisasjonsendring. **Bimodal IT blir altså brukt for å kombinere kortsiktig og langsiktig utvikling.** Utfordringen er å integrere de to modusene.

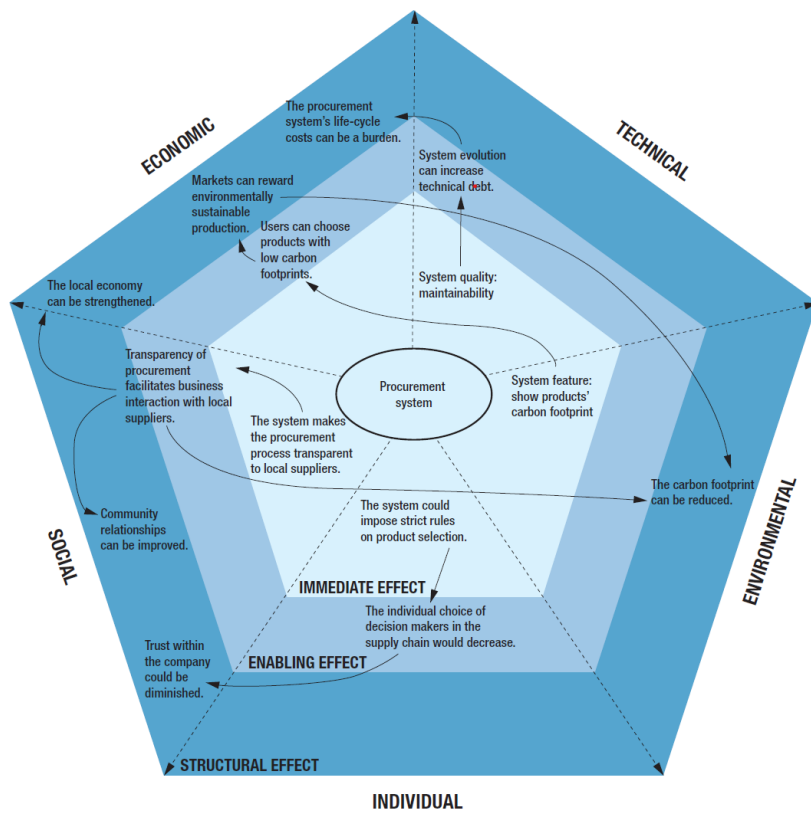
	Mode 1	Mode 2
Goal	Reliability	Agility
Value	Price for performance	Revenue, brand, customer experience
Approach	Waterfall, V-model, "high-ceremony IID"	Agile, Kanban, "low-ceremony IID"
Governance	Plan-driven, approval-based	Empirical, continuous, process-based
Sourcing	Enterprise suppliers, long-term deals	Small, new vendors; short-term deals
Talent	Good for conventional processes and projects	Good for new and uncertain projects
Culture	IT-centric, removed from customer	Business-centric, close to customer
Cycle times	Long (months)	Short (days, weeks)

Kortsiktig vs. langsiktig modus

Den kortsiktige, hurtige og smidige modus 2 er nødvendig for at organisasjonen skal kunne fremme innovasjon og lettere håndtere endringer og usikkerhet.

Den langsiktige og sekvensielle modus 1 er nødvendig for å utvikle sikre, pålitelige og trygge systemer som krever mye planlegging og struktur. Det støtter også bærekraftig utvikling mht. faktorer innenfor miljø, økonomi, sosial og individuelle omgivelser og teknologi. Som vi kan se på figuren, kan påvirkningen til programvaren deles inn i:

- **Umiddelbar effekt** = direkte effekt på produksjonen, bruk og kasting av programvaresystem
- **Muliggjørende effekt** = oppstår over tid
- **Strukturell effekt** = vedvarende endringer som kan observeres på makronivå



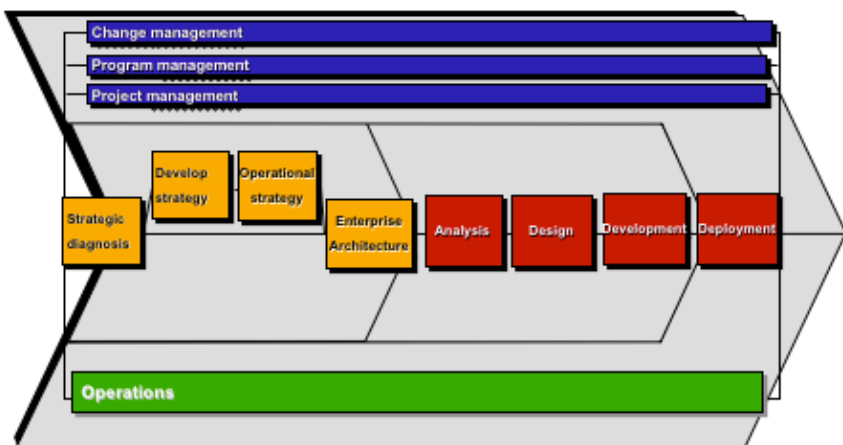
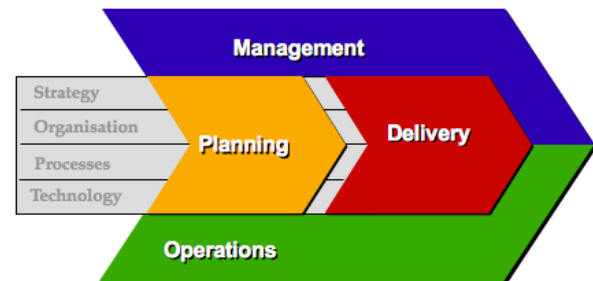
DevOps

DevOps er en praksis der drifts- og utviklingsteamet samarbeider i løpet av hele livssyklusen, fra design, via utviklingsprosessen til produksjonsstøtte. Det er en prosess med integrert utvikling og drift, noe som støtter raske utgivelser.

BI metodikk

Hovedkomponentene til BI metodikken (se figur) utgjør til sammen den fullstendige livssyklusen til en endringsprosess, og de representerer det høyeste nivået i strukturen til BI metodikken. Hovedkomponentene er:

- **Planlegging** = de viktigste business mulighetene til organisasjonen blir identifisert og det formuleres en plan for hvordan disse kan best mulig utnyttes
- **Endringsledelser** = gir en kontinuerlig oversikt, noe som sikrer føring, fremdrift og verdiskapning i endringsprosessen
- **Levering/gjennomføring** = fokuserer på å planlegge og gjennomføre utviklingen av nye verdiskapende business funksjoner.
- **Drift og kontinuerlig forbedring** = fokuserer på fortløpende drift og forbedring for å innfri fastsatte strategier og målsetninger



Figuren viser hvordan hovedkomponentene kan deles opp. Planleggingen og leveringingen kan deles inn i etapper som utføres etter hverandre, mens ledelse og drift deles inn i prosesser som er kontinuerlige aktiviteter over tid.

Følgende er eksempler på hva de ulike hovedkomponentene kan innebære ved en endringsprosess:

- **Planlegging** = det blir laget et overordnet situasjonsbilde og en enighet om hvilke tiltak som gir størst gevinst, noe som er viktig i starten av endringsprosessen
- **Endringsledelser** = det blir gitt overordnet føring og en detaljert fremdriftskontroll fra starten av prosessen til slutten.
- **Levering/gjennomføring** = en identifisert endring blir planlagt og gjennomført
- **Drift og kontinuerlig forbedring** = endringstiltak videreføres og forbedres over tid for å sikre langvarig suksess og verdiskapning

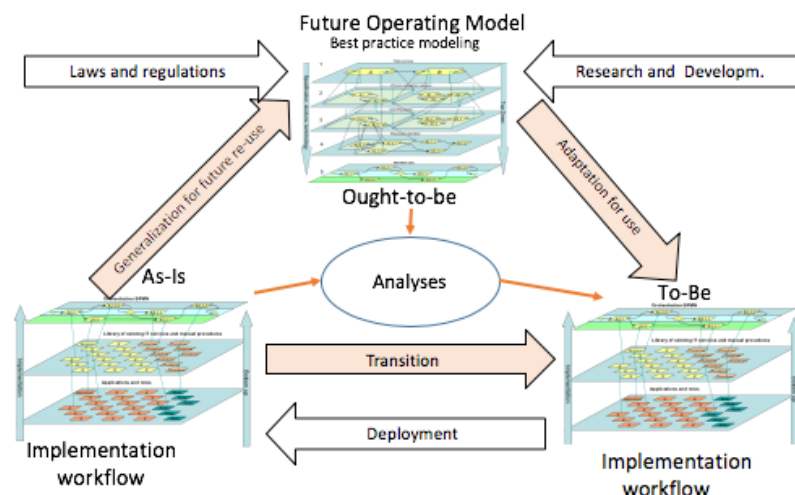
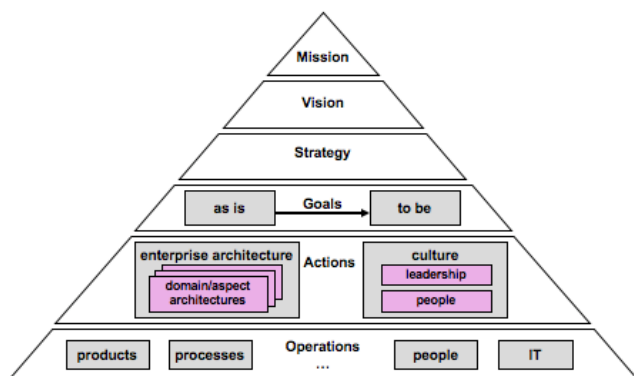
Enterprise arkitektur

De fleste metodikkene vi har sett på frem til nå har fokusert på utvikling og evolusjon av IT-system, mens andre metodikker har en bredere tilnærming og ser på hele bedriften (*enterprise*). Disse metodikkene kalles **Enterprise Arkitektur**, og de ble introdusert for 25 år siden for å adressere to problemer:

- **Komplekse systemer** = organisasjoner brukte stadig mer penger på å bygge IT systemer
- **Dårlig forretningsinnretning** = det ble stadig vanskeligere for organisasjoner å innrette de dyre IT systemene med deres forretningsbehov

Utvikling av IT systemer ble altså **mer kostbart og ga mindre verdi**. Strategien for IT og forretningen var separert, så **Enterprise Arkitektur ble introdusert som en bru mellom forretningsstrategi og implementasjon**. Forretningsstrategi involverer drivere, retningslinjer, trendanalyse og mål, mens implementasjon innebærer applikasjonssystem, teknologisk infrastruktur, organisasjonsstruktur og business prosesser. Dermed er det fokus på andre faktorer enn bare implementasjon av IT.

Enterprise Arkitektur er basert på analyser av AS-IS, TO-BE og Ought-to-be situasjonene i organisasjonen. Her vil AS-IS representere slik situasjonen er i dag, Ought-to-be representerer modellen for beste praksis, mens TO-BE representerer den fremtidige foreslåtte løsningen. Den beste praksisen blir tilpasset bruken.









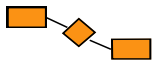
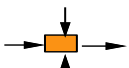
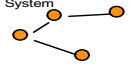
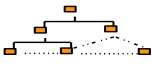
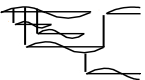
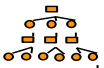
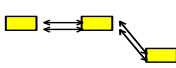
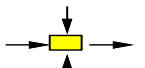
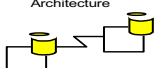
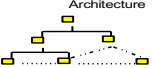

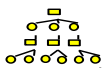
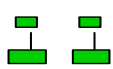











AS-IS og TO-BE brukes for å illustrere målene ved en overgang, mens Enterprise Arkitektur gir handlingene som trengs for å utføre overgangen (se figur).

Organisasjoner bruker Enterprise Arkitektur for å optimalisere de fragmenterte prosessene på tvers av hele bedriften. **Hovedgrunnen til å bruke en EA er derfor at det gir en oversikt over virksomhetens prosesser, systemer, teknologier, strukturer og evner. Dette gir en strategisk kontekst for IT systemene som må utvikles som følge av de konstante endringene i virksomhetens behov.** For at man skal kunne utvikle IT systemene slik at de

håndterer fremtidige endringer, er det viktig at man kjenner konteksten eller omgivelsene. Dette gjør at IT-systemene ikke blir utviklet i et vakuum, men heller som en del av et digitalt økosystem. Forskjellen mellom EA og smidig utvikling, er at EA er fokusert på prosessen, mens smidig utvikling er fokusert på produktet. Enterprise Arkitektur kan bli veldig omfattende, så det er viktig å bruke riktig nivå av abstraksjon eller kombinere med smidig utvikling. Et alternativ er Bimodal IT, som også tar hensyn til forretningsdelen ved utviklingen av IT-system.

Zachman rammeverk

Zachman er et rammeverk som gir en formell måte å definere en bedrift (*enterprise*). Den er uavhengig av prosessene eller verktøyene som brukes for å produsere beskrivelsen av bedriften. Rammeverket består av en matrise der kolonnene viser fokusområdene eller aspektene ved en bedrift og radene viser detaljering eller abstraksjonsnivå

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>
SCOPE (CONTEXTUAL) <i>Planner</i>	List of Things Important to the Business 	List of Processes the Business Performs 	List of Locations in which the Business Operates 	List of Organizations Important to the Business 	List of Events Significant to the Business 	List of Business Goals/Strat 
ENTERPRISE MODEL (CONCEPTUAL) <i>Owner</i>	e.g. Semantic Model  Ent = Business Entity Rein = Business Relationship	e.g. Business Process Model  Proc. = Business Process I/O = Business Resources	e.g. Business Logistics System  Node = Business Location Link = Business Linkage	e.g. Work Flow Model  People = Organization Unit Work = Work Product	e.g. Master Schedule  Time = Business Event Cycle = Business Cycle	e.g. Business Plan  End = Business Objective Means = Business Strategy
SYSTEM MODEL (LOGICAL) <i>Designer</i>	e.g. Logical Data Model  Ent = Data Entity Rein = Data Relationship	e.g. Application Architecture  Proc. = Application Function I/O = User Views	e.g. Distributed System Architecture  Node = I/S Function (Processor, Storage, etc.) Link = Line Characteristics	e.g. Human Interface Architecture  People = Role Work = Deliverable	e.g. Processing Structure  Time = System Event Cycle = Processing Cycle	e.g. Business Rule Model  End = Structural Assertion Means = Action Assertion
TECHNOLOGY MODEL (PHYSICAL) <i>Builder</i>	e.g. Physical Data Model  Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	e.g. System Design  Proc. = Computer Function I/O = Data Elements/Sets	e.g. Technology Architecture  Node = Hardware/System Software Link = Line Specifications	e.g. Presentation Architecture  People = User Work = Screen Format	e.g. Control Structure  Time = Execute Cycle = Component Cycle	e.g. Rule Design  End = Condition Means = Action
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT) <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Stmt I/O = Control Block	e.g. Network Architecture  Node = Addresses Link = Protocols	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle	e.g. Rule Specification  End = Sub-condition Means = Step
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY

Begge dimensjonene er viktige for å fullstendig forstå bedriften. De ulike aspektene (kolonnene) ved bedriften er:

- **Data (hva)** = ser på dataen som trengs for at bedriften skal operere
- **Funksjon (hvordan)** = ser på hvordan bedriften opererer
- **Nettverk (hvor)** = ser på hvordan bedriftens aktiviteter er geografisk fordelt
- **Mennesker (hvem)** = ser på menneskene som utfører arbeidet, tildeling av arbeid og relasjoner mellom mennesker
- **Tid (når)** = ser på forholdet mellom hendelser som utgjør resultatkriteriet
- **Motivasjon (hvorfor)** = ser på motivasjonen til bedriften, og fokuserer på mål

De ulike detaljeringene/abstraksjonsnivåene (radene) er:

- **Omfang (kontekst)** = gir tilnærming på størrelse, form, relasjoner og grunnleggende formål ved den endelige strukturen
- **Enterprise modell (konseptuell)** = designet av bedriften eller arkitektens tegning
- **System modell (logisk)** = designet oversettes til detaljerte spesifikasjoner.
- **Teknologi model (fysisk)** = arkitektens modell oversettes til byggerens modell

- **Detaljert representasjon (detaljert)** = detaljerte spesifikasjoner blir gitt til programmerere

I bunnen har man et funksjonell bedrift der et system er implementert og gjort en del av bedriften (enterprise). Zachman rammeverket er ingen metodikk, fordi den gir ikke spesifikke metoder som kan brukes for å samle eller styre informasjonen den beskriver. Fordeler med Zachman rammeverket er at det er enkelt å forstå, det er omfattende (beskriver hele bedriften), det gir et språk som kan brukes for å beskrive komplekse konsepter på enkle måter, det hjelper til i beslutningstakingen (valg blir ikke tatt i vakuum) og det er nøytralt (uavhengig av verktøy og metodologier).

TOGAF

TOGAF (The Open Group Architecture Framework) ble introdusert i 1997 av Open Group som er en internasjonal interesseorganisasjon. **TOGAF er basert på en ADM (Architectre Development Methodology) som er en syklus som består av en rekke steg** (se figur). Dette inkluderer:

1. **Business arkitektur** = beskriver prosessene bedriften bruker for å oppnå sine mål
2. **Applikasjon arkitektur** = beskriver hvordan bestemte applikasjoner er designet og interagerer med hverandre
3. **Data arkitektur** = beskriver hvordan bedriften organiserer og aksesserer datalagrene
4. **Teknisk arkitektur** = beskriver maskinvaren og programvaren som støtter applikasjoner og deres interaksjon



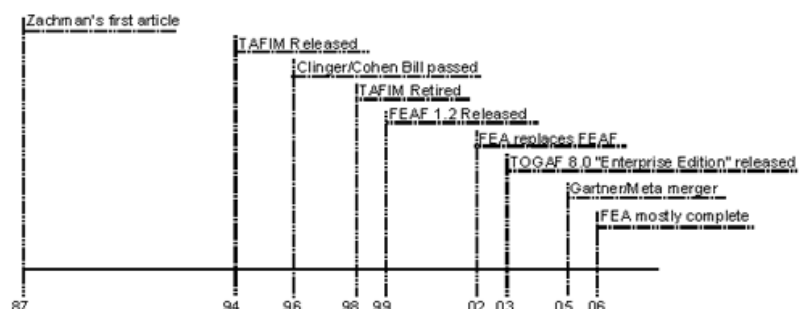
Fordelen ved TOGAF er at den er fleksibel når det gjelder arkitekturen som lages. Ulemper er at det er en omfattende prosess (fra forretningskrav til applikasjon til infrastruktur) og den endelige arkitekturen kan være god eller dårlig, siden TOGAF kun beskriver hvordan man lager EA og ikke hvordan man lager en god arkitektur.

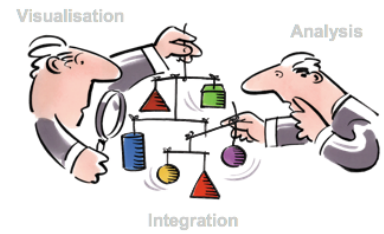
Framvoksende og smidig EA (emergent and agile)

Når en Enterprise Arkitektur utvikles vil det som regel brukes både tekstlige og grafiske representasjoner som består av et sett med modeller som blir presentert i en av hovedverktøyene til EA. Disse modellene brukes av beslutningstakere i organisasjonen for å se effekten av virkelige eller foreslåtte endringer. En avgjørelse eller endring kan ha signifikant innvirkning på teknologi, data og applikasjoner, noe som kan resultere i uforventede konsekvenser og kostnader. Utfordringer ved produksjon av tradisjonelle EAs er at globalisering gjør at arbeidskraften er spredt over flere lokasjoner, det er store endringer i arbeidsprosesser innenfor interaksjon, samarbeid, arbeidsmiljø og teknologi, IT prosjekt blir stadig større og mer komplekse, osv. **Et hovedproblem ved tradisjonelle enterprise arkitekturer er at de er statiske og tar ikke hensyn til at bedrifter er fremvoksende (emergent), komplekse og tilpassende systemer som krever en smidig tilnærming.**

Emergent og smidig EA gjenkjenner organisasjoner som komplekse og tilpassningsdyktige systemer under konstant endring.

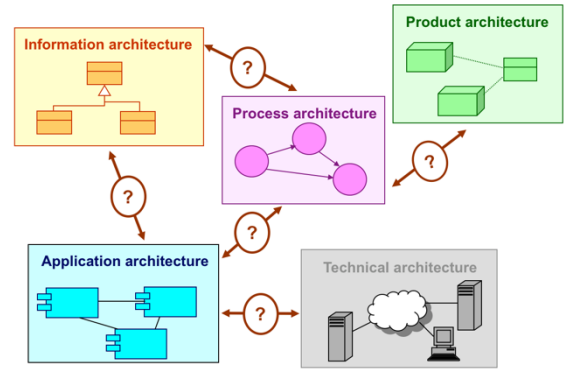
Gartner gruppen foreslo en ny tilnærming til EA, der det legges mer til rette for innovasjon, arkitektene forstår at de ikke kan kontrollere alle aspektene ved arkitekturen, systemet må kunne respondere på endringer, osv.



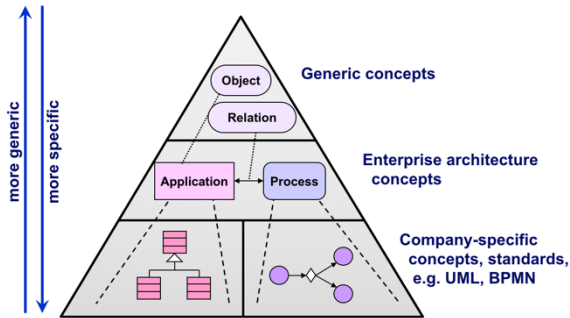


ArchiMate – modellering av EA

ArchiMate (Architecture-Animate) er et EA modelleringsspråk som støtter integrering, analyse og visualisering av arkitekturer på en entydig måte. Det brukes for å beskrive konstruksjonen og operasjonen av business prosesser, organisasjonsstrukturer, informasjonsflyt, IT-systemer og teknologisk infrastruktur. Det brukes også for å beskrive sammenhengen mellom ulike deler av arkitekturen (se figur). Dette kan hjelpe interessenter til å identifisere, vurdere og kommunisere konsekvenser av avgjørelser og endringer innenfor disse områdene. **ArchiMate gir bedre støtte til en enterprise arkitekt**, siden det gir **presis dokumentasjon** av EA-nivået (kan integreres andre språk som UML og BPMN) og kan brukes for å **kommunisere arkitekturen** med andre og **analysere arkitekturen** før den implementeres. ArchiMate tilfredsstillers behovet for en velbegrunnet, praktisk og leverandøruavhengig standard for EA modellering. Fordeler med ArchiMate er:



- **Magert språk** med akkurat nok konsepter (inkluderer ikke «alt»)
- **Velbegrunnede konsepter og modeller** gir presis og tydelig kommunikasjon
- **Koblet til eksisterende tilnæringer** (eks: UML og BPMN)
- **Internasjonal og leverandøruavhengig standard**
- **Støtter flere verktøy**

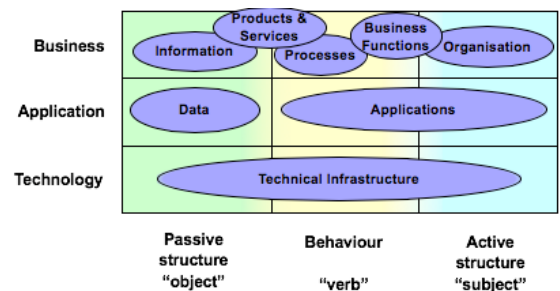


ArchiMate modellerer EA konsepter som er mer generell og fokuserer på hele bedriften, til forskjell fra UML og BPMN som er mer spesifikke.

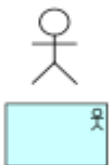
ArchiMate rammeverket deler enterprise arkitekturen inn i tre lag:

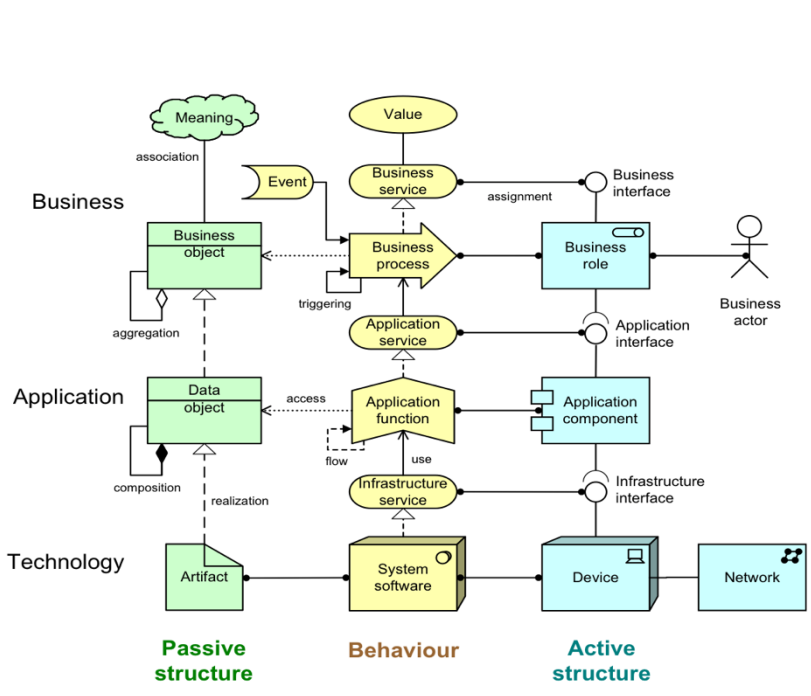
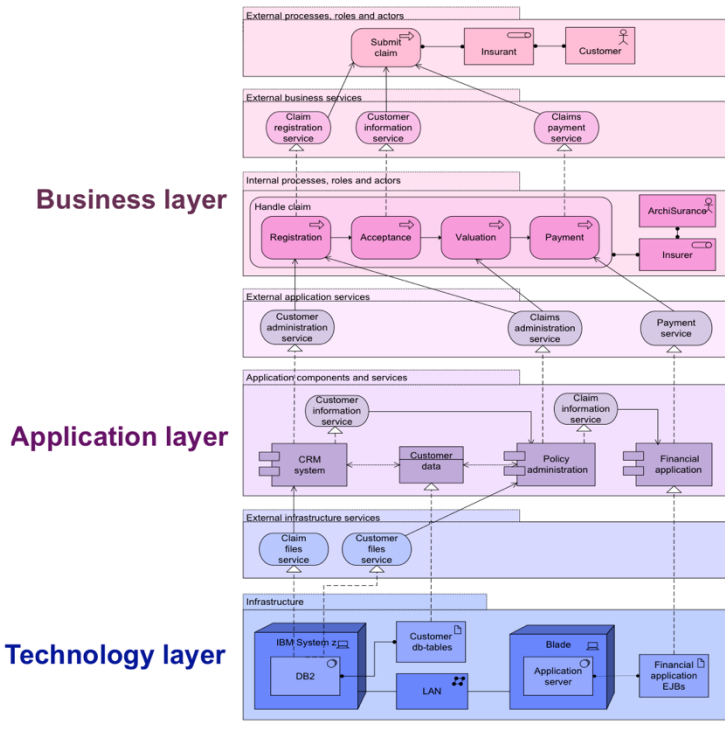
1. **Business lag** = informasjon om bedriften, produkter og tjenester bedriften tilbyr, prosesser, business funksjoner og organisasjoner
2. **Applikasjonslag** = data og applikasjoner i bedriften
3. **Teknologi lag** = teknologisk infrastruktur

Ved hvert blir igjen delt inn i **passive strukturer** (objekter), **oppførsel** (verb) og **aktive strukturer** (subjekt).



I ArchiMate notasjon blir ikoner eller bokser med ikoner brukt for å representere de fleste konseptene. Elementer med skarpe hjørner er strukturer, mens elementer med avrundede hjørner er oppførsel. Notasjonen ligner UML og BPMN, slik at det er lettere å lære for bedriften. Relasjoner blir representert med piler og er også likt andre språk. Figuren til høyre på neste side viser et eksempel på en ArchiMate modell.





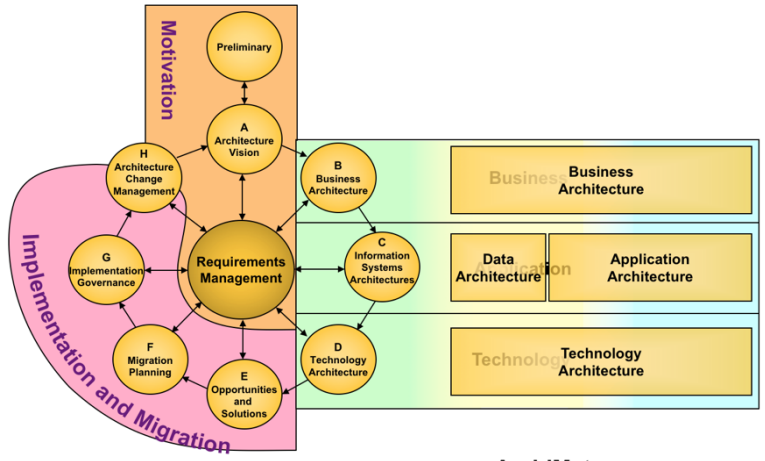
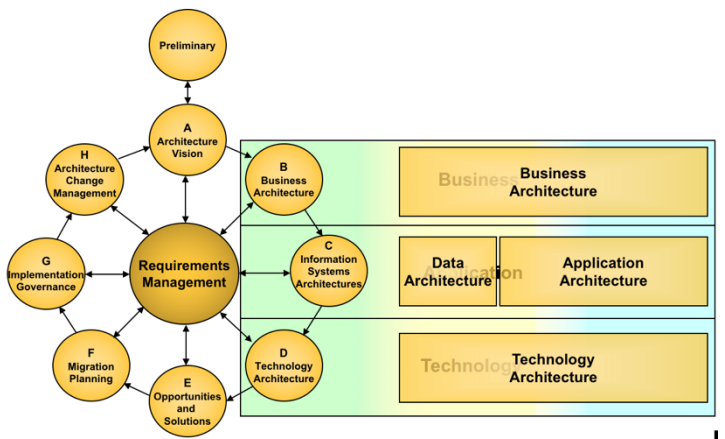
ArchiMate kan brukes for å modellere enterprise arkitekturer laget med Zachman rammeverk eller TOGAF (se figurer under). TOGAF bruker ADM for å gi metoder og måte å arbeide, men gir ingen beskrivende teknikk. ArchiMate er derimot et beskrivende språk, men gir ingen metoder eller måter å arbeide. Dvs. de komplementerer hverandre.

	What	How	Where	Who	When	Why	
Scope = Planner's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	Contextual
Enterprise Model = Owner's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	Conceptual
System Model = Designer's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	Logical
Technology Model = Builder's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	Physical
Detailed representation = Subcontractor's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	As Built
Functioning Enterprise = User's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	Functioning

Zachman rammeverket

	What	How	Where	Who	When	Why
Scope = Planner's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]
Enterprise Model = Owner's view	Information	Business	Structure		Attributes	Motivation extension
System Model = Designer's view	Information	Application	Structure		Attributes	Motivation extension
Technology Model = Builder's view	Information	Technology	Structure		Attributes	Motivation extension
Detailed representation = Subcontractor's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]
Functioning Enterprise = User's view	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]	[Icon]

Forholdet mellom Zachman rammeverket og ArchiMate modelleringen. Legg merke til at kolonnene for når og hvorfor er utenfor ArchiMate og dekkes av utvidelsene som kalles attributter og motivasjon.



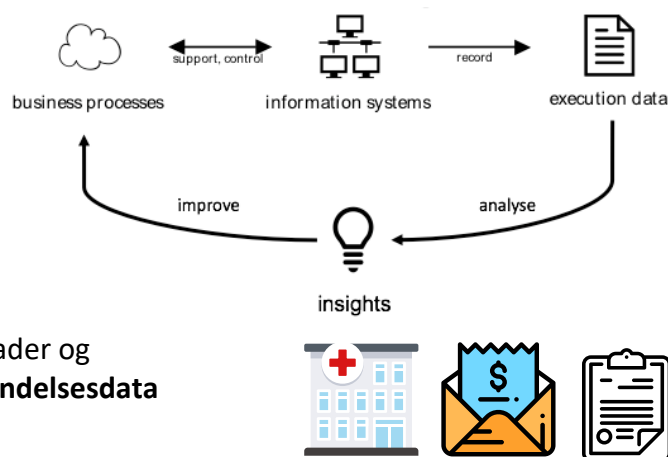
Her kan vi se hvordan tre av stegene i ADM syklusen til TOGAF kan deles inn i de tre lagene: business, applikasjon og teknologi arkitektur

De resterende stegene i TOGAF er ansvarlig for motivasjon, implementering og migrering.

Introduksjon til prosessmining

Prosessmining er teknikker som prøver å forbedre prosesser (ofte business prosesser) ved å analysere hendelses (*execution data*) fra hendelseslogger i systemet. De brukes for å oppdage, følge med på og forbedre virkelige prosesser ved å hente kunnskap fra tilgjengelige hendelseslogger i systemet som lagrer nåværende informasjon om organisasjonen.

De vanlige komponentene i prosessmining er **business prosesser** (eks: fakturabehandling, lånesøknader og pasientlogistikk), **IT system** (støtter prosessene) og **hendelsesdata** (*execution data*).



Ved BPR (Business Process Reengineering) er organisasjoner som regel mest interessert i å forbedre TO-BE prosessen, men forståelse av den nåværende prosessen er essensielt for å:

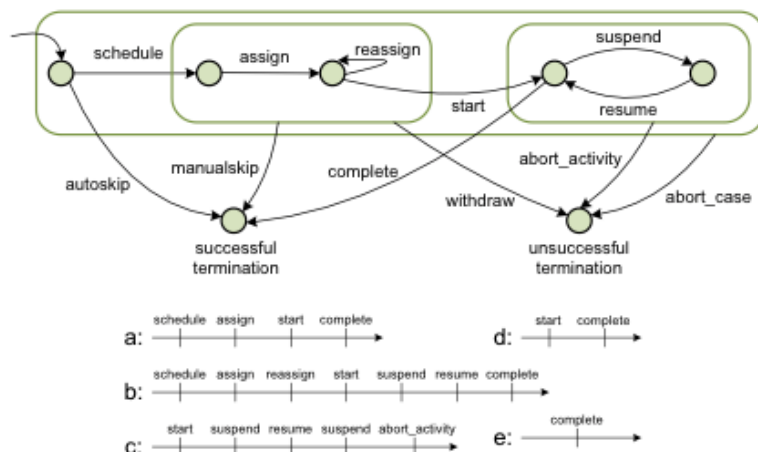
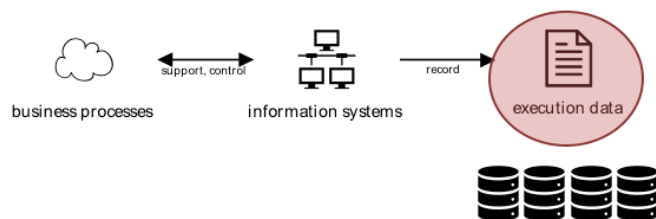
1. Vite om det er **verdt å investere i forbedringene**
2. Identifisere **hvor ytelsesproblemet ligger**
3. Vite **hvor mye variasjon det er i prosessen** på tvers av organisasjonen

Bedrifter har en tendens til å hoppe over prosessanalysen, bruke snarveier eller betale konsulenter for at de skal analysere AS-IS prosessen. **En av de største utfordringene ved digital transformasjon innenfor en organisasjon er å forstå hvordan prosesser faktisk utføres.** Når man designer prosesser kan man forvente at de vil utføres i henhold til designet, men designet er ofte ufullstendig, slik at prosessene utføres på en annen måte i praksis. **Prosessmining gir et sett med verktøy og teknikker som kan brukes for å analysere business prosesser. Prosessmining bruker prosess hendelsesdata (*execution data*) som er tilgjengelig i det underliggende informasjonssystemet og kartlegger og visualiserer hvordan prosessene blir utført i virkeligheten.** Dette vil fjerne alle antagelser og gjetninger om prosessene.

Hva er en prosess hendelsesdata?

For å forstå prosess hendelsesdata begynner vi med å definere to begrep:

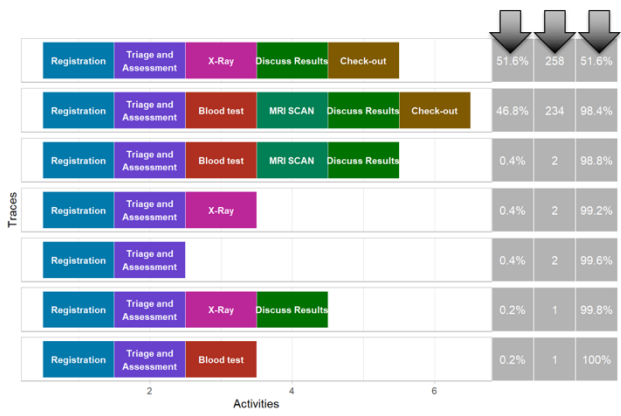
- **Prosess** = et sett med sammenhengende eller interagerende aktiviteter, som forvandler input til output
- **Hendelse** = noe som har skjedd ved et spesifikt tidspunkt. For å definere det mer formelt må vi se på begrepet aktivitet. Figuren viser livssyklusene til flere aktiviteter innenfor samme prosess, for eksempel innebærer aktivitet a fire steg. Det kan være flere instanser av en aktivitet, og disse kan ha ulike lengde (se figur). En hendelse indikerer at det har skjedd en livssyklus overgang for en bestemt aktivitetsinstans i konteksten til en bestemt case, ved et spesifikt tidspunkt.



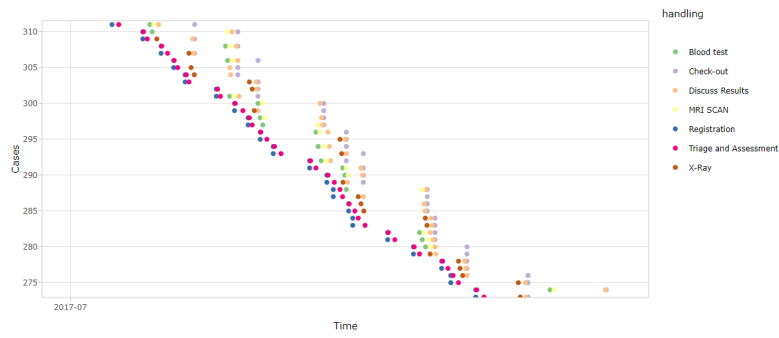
En prosess består av flere aktiviteter som består av flere aktivitetsinstanser, der hver instans utgjør en hendelse. En hendelseslogg kan ses på som en tabell som lagrer hendelser, i form av aktivitet, case, instans, tidspunkt, osv. (se figur).

Activity	Case	Resource	Activity Instance	Lifecycle Transition	Timestamp	
<handling <fctr>	<patient <chr>	<employee <fctr>	<handling_id <chr>	<registration_type <fctr>	<time <S3: POSIXct>	<order <int>
Registration	1	r1	1	start	2017-01-02 11:41:53	1
Registration	2	r1	2	start	2017-01-02 11:41:53	2
Triage and Assessment	1	r2	501	start	2017-01-02 12:40:20	501
Registration	1	r1	1	complete	2017-01-02 12:40:20	2722
Registration	2	r1	2	complete	2017-01-02 15:16:38	2723
Triage and Assessment	2	r2	502	start	2017-01-02 22:32:25	502
Triage and Assessment	1	r2	501	complete	2017-01-02 22:32:25	3222
Triage and Assessment	2	r2	502	complete	2017-01-03 12:34:01	3223
Registration	4	r1	4	start	2017-01-04 01:34:04	4
Registration	3	r1	3	start	2017-01-04 01:34:05	3

1-10 of 5,442 rows Previous 1 2 3 4 5 6 ... 545 Next



En hendelseslogg kan også være en registrering av aktivitetssekvenser (figur til venstre) eller tidsserier (figur under).



Typer prosessmining

Vi ser nærmere på to hovedtyper av prosessmining.

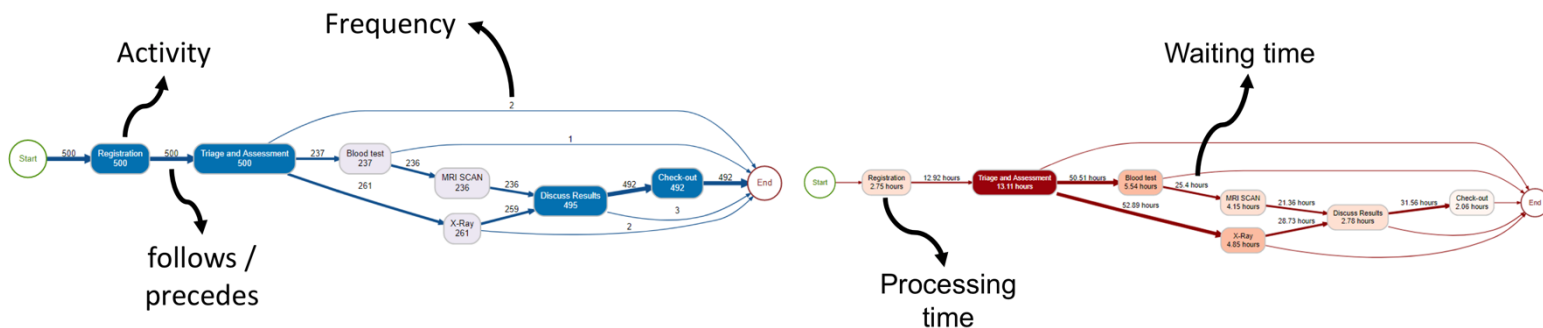
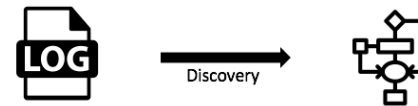
Conformance Checking (samsvarstest)

Conformance Checking brukes for å se om det er samsvar mellom en prosessmodell og hendelsesloggen til samme prosess. For eksempel kan prosessmodellen gi at kjøpsordre på mer enn en million krever to sjekker, og analyse av hendelsesloggen viser om denne regelen følges eller ikke. **Conformance teknikkene tar prosessmodellen og hendelsesloggen som input og returnerer forskjellene mellom oppførselen fanget i prosessmodellen og oppførselen fanget i hendelsesloggen.** Disse forskjellene kan representeres visuelt ved å legges over prosessmodellen eller tekstuert ved å gis som en liste.

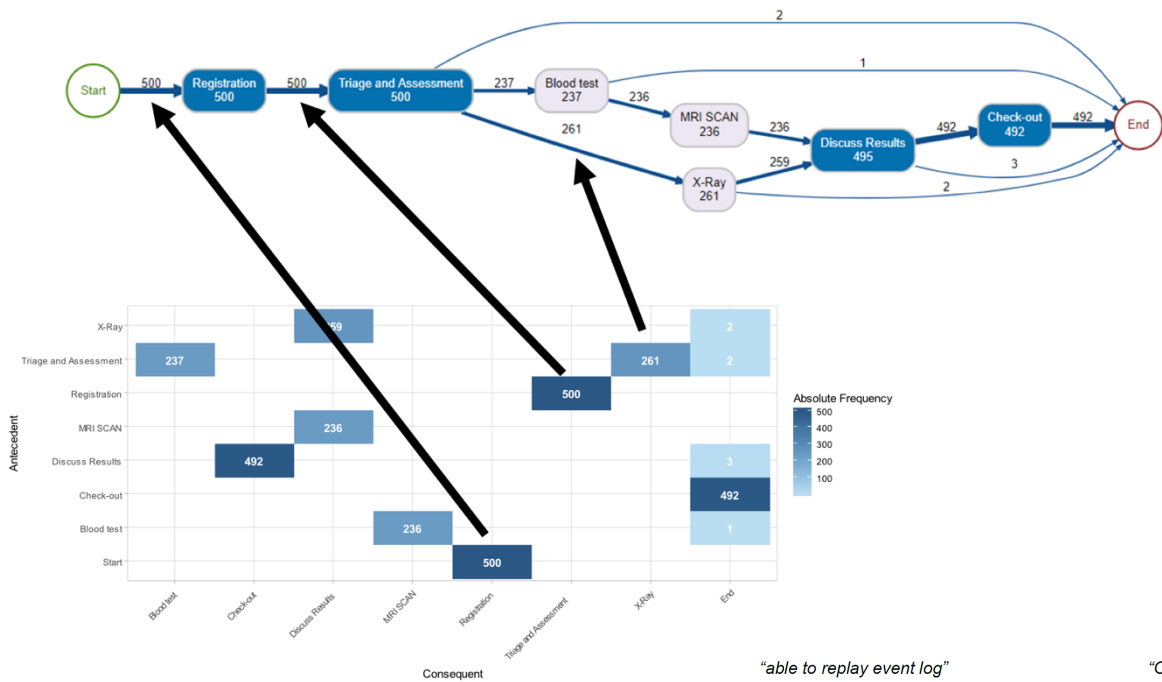


Business Process Discovery (BPD)

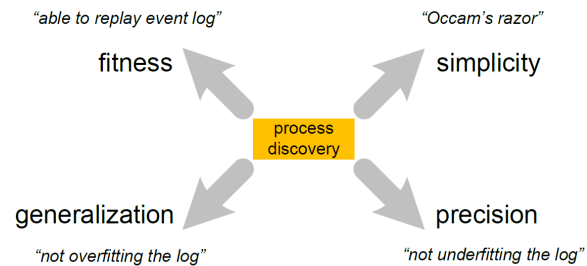
Business Process Discovery (BPD) brukes for å lage en representasjon av organisasjonens nåværende business prosesser og viktige prosessvariasjoner. BPD tar hendelsesloggen som input og returnerer en prosessmodell. Det brukes for å tegne **prosesskartet**, som gir aktivitetene i prosessen og deres frekvens, prosesseringstid, ventetid og rekkefølge (se figurer under).



I et prosesseringskart blir hver aktivitet representert av en node, og det er også vanlig å inkludere noder for starten og slutten av prosessen. Relasjoner mellom aktiviteter representeres av buer/kanter. En bue fra aktivitet A til aktivitet B, betyr at aktivitet A blir fulgt av aktivitet B minst en gang. Buer kan betegnes med blant annet frekvens og ventetid. De kan også tegnes som grafer, som vi kan se på figuren under.



Utfordringer ved tegning av prosesskart er at parallelle aktiviteter, overgeneralisering og sjelden oppførsel kan gi komplekse kart som fører til misforståelser av prosessen. For å sikre at modellen har en høy kvalitet, må man bruke en god balanse mellom **presisjon, generalisering, enkelhet og hvor godt det passer.**



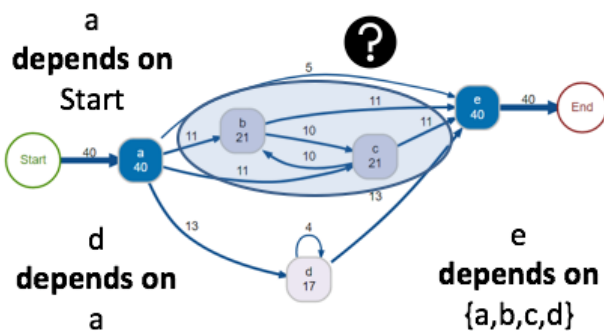
$$a \Rightarrow^L b = \begin{cases} \frac{|a >^L b| - |b >^L a|}{|a >^L b| + |b >^L a| + 1}, & \text{for } a \neq b \\ \frac{|a >^L a|}{|a >^L a| + 1} & \text{otherwise} \end{cases}$$

Intuition

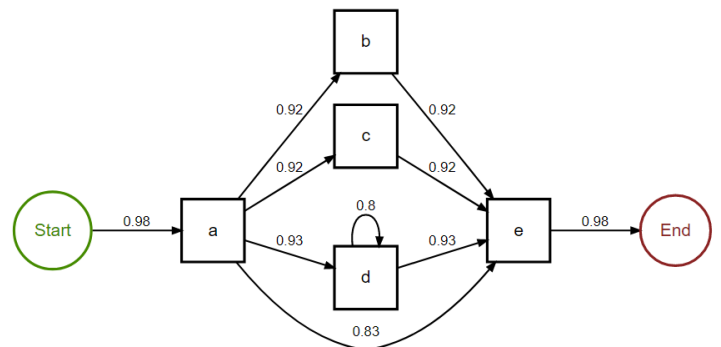
- $|a \Rightarrow^L b|$ goes towards 1 Causal: b depends on a
- $|a \Rightarrow^L b|$ goes towards 0 Parallel: No dependency
- $|a \Rightarrow^L b|$ goes towards -1 Inverse Causal: a depends on b

En annen måte å håndtere disse utfordringene er å lage modeller som fanger avhengigheter mellom aktiviteter istedenfor relasjoner. Dette gjøres ved å regne ut avhengighetsmålet som gis av formelen på figuren.

L Event log
a and **b** Activities / activity identifiers
 $a >^L b$ "b directly-follows a" or "a directly-precedes b" in L
| Frequency / Set cardinality

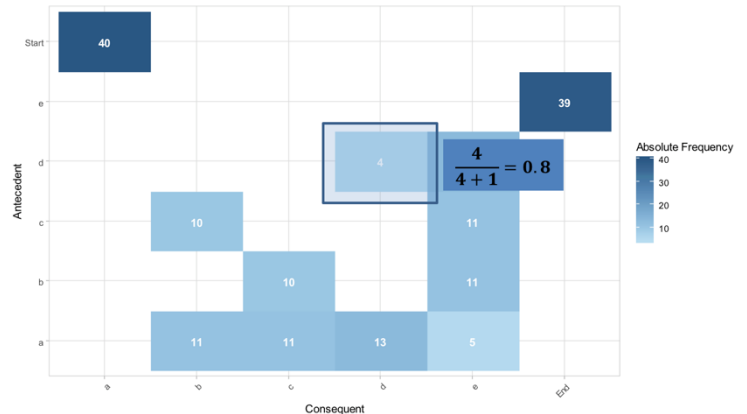
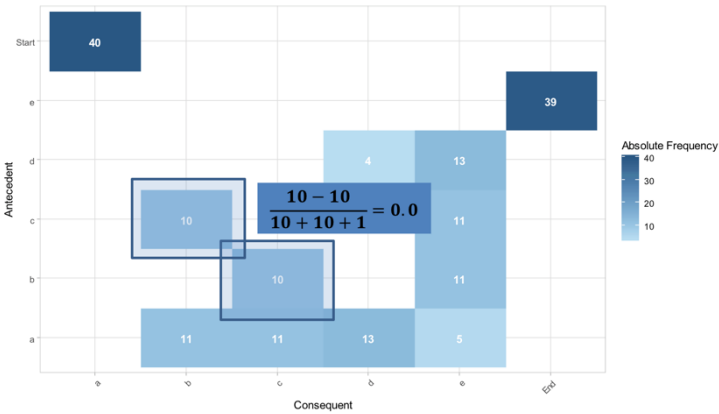
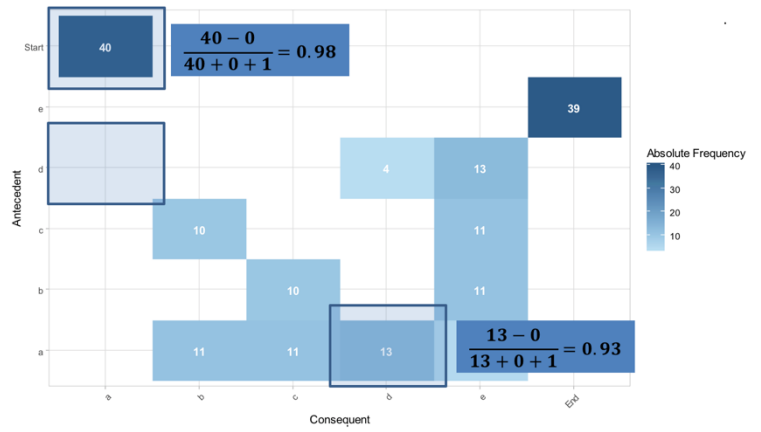
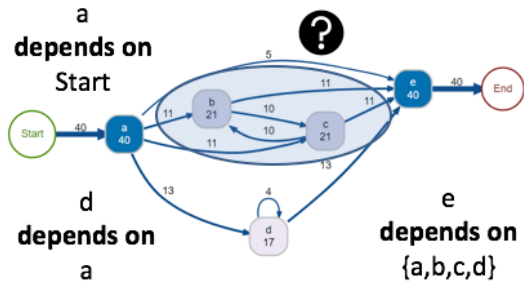


Prosesskart som fanger relasjonene mellom aktivitetene

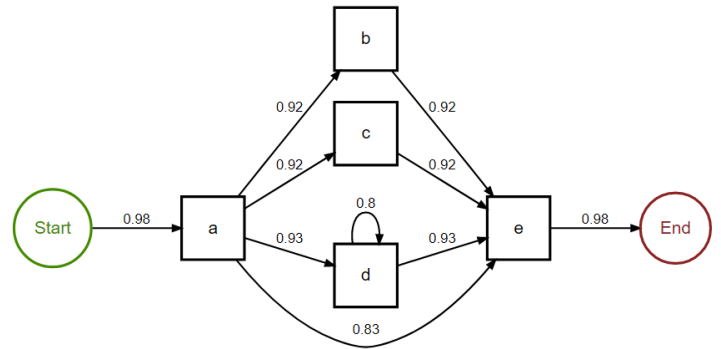
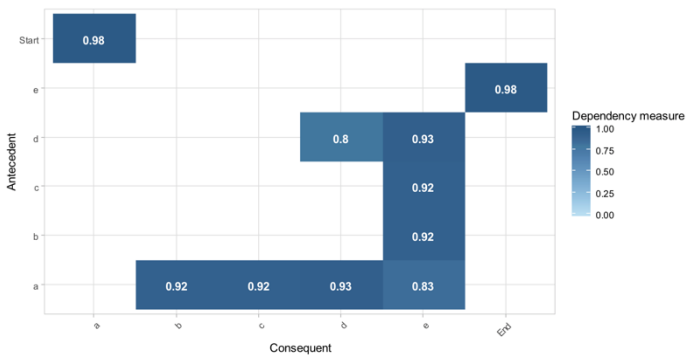


Prosesskart som fanger avhengighetene mellom aktivitetene

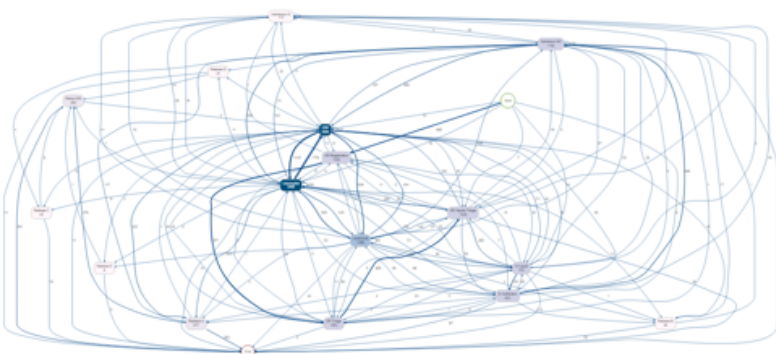
Figurene på neste side viser overgangen mellom disse.



Figurene over viser hvordan vi regner ut avhengighetene for de ulike relasjonene, ved å bruke frekvensen til relasjonene og formlene på forrige side. Figurene under viser resultatet.



Ved å bruke avhengigheter kan vi gjøre prosesskartet mer forståelig i tilfeller der det er mange parallelle baner, mye generalisering og flere sjeldne oppførsler:

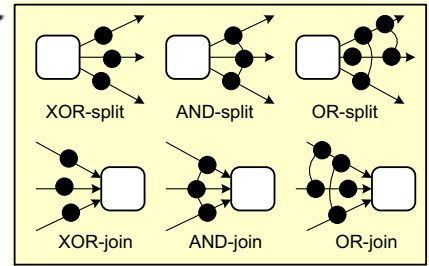
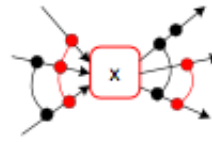


Prosesskart med relasjoner



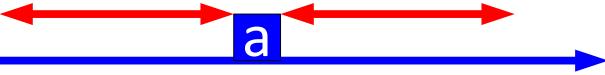
Prosesskart med avhengigheter

Når man lager prosesskart må man også bruke **input og output bindinger** som lager forgreininger eller slår sammen forgreininger i modellen. Vi skiller mellom **XOR-, AND- og OR-splitt** som deler opp relasjonen/avhengigheten i flere baner, og **XOR-, AND- og OR-join** som slår sammen flere forgreininger (se figurer). Det kan være utfordrende å bruke gateways på riktig måte.



scan trace in window before a and guess what the input binding was based on observed activities

scan trace in window before a and guess what the output binding was based on observed activities



For å bestemme hvilken type gateway man skal bruke kan man se på tidsvinduet før og etter hver aktivitet og telle antall input og output aktiviteter (se figur). Dette kan gjøres vha en grådig

søkestrategi som tar utgangspunkt i en aktivitet og ser på hvilke aktiviteter som er input og output ettersom prosessen gjennomføres. På figuren kan vi se et eksempel der søket tar utgangspunkt i aktivitet a. Den grådige algoritmen brukes for å finne ut at startaktiviteten er input 40 ganger, b-aktiviteten er output 1 gang, aktivitetssekvensen b-c eller c-b er output 20 ganger, aktivitet d er output 13 ganger og aktivitet e er output 5 ganger. Sekvensen med b- og c-aktivitetene modelleres som en AND-splitt, mens resten modelleres som XOR-splitt. Dette gjentas for de andre aktivitetene, og vi får prosesskartet under:

