Kungl Tekniska Högskolan/Stockholms Universitet
Institutionen för Informationsbehandling - ADB
1971-11-08

Utvärdering av databehandlingssystem

Tomas Ohlin

Framarbetat med stöd från Statens Naturvetenskapliga Forskningsråd dnr 2861

Inst för informationsbehandling - ADB
T Ohlin
Arbetspapper maj 1971
4 sidor

Definition av problem i samband med utvärdering av datorsystem

Vid angripande av problemen kring utvärdering av datorsystem framträder det faktum att den maskinella utrustningen endast utgör ett delsystem inom informationssystemet. Samspelet människa-maskin tilldrar sig intresse, och det är den samlade effekten av användare och datorsystem som primärt kräver studium. Det är detta samlade system som främst skall betjäna informationssystemet.

Den prestandaanalys som skall leverera information till utvärderingsprocessen avser sålunda prestanda för hela systemet: användare plus dator.

I själva verket innebär utvärderingsarbetet som en viktig del att arbeta fram beslutsinformation rörande om dator(er) över huvud taget skall anskaffas eller förnyas. Det kan motiveras med införande av ett datorsystem inom ett företags informationssystem endast om det kan visas (eller åtminstone göras tillräckligt troligt) att detta, totalt sett, positivt bidrar till företagets måluppfyllelse. Ställningstagande rörande datoranskaffning kan sålunda knappast ses som ett i företaget isolerat och rent datatekniskt beslut.

Utgående från det ovan nämnda listas nedan problem som är av betydelse för utvärderingen. Problemen är i vissa fall delvis överlappande. Listan kan givetvis göras betydligt längre, d v s nedbrytas ytter ligare, vilket kommer att ske under arbetets gång .

1. Vid vilken nedbrytningsnivå ("på vilket djup") skall informationsmängder och processer befinna sig för att utgöra bästa bas för syntesarbetet (inkl utrustningsanpassning)?
2. Hur skall informationsmängder och processer syntetiseras över den ur analysfasen erhållna nedbrytningsnivån, d v s vilka kriterier skall användas?
3. Vilken arbetsgång är lämplig i syntesen för att minimera resursåtgången för syntesarbetet under beaktande av den önskade ambitionsnivån för det färdiga systemet?
4. Är resursåtgången i syntesarbetet positivt korrelerad till antalet erforderliga iterationer i arbetet?
5. På vilket sätt är syntesarbetets kvalitativa resultat avhängigt resursåtgången?
6. Kan företagets "behov" av datorkapacitet på förhand anges "in blanco"?
7. Med vilken säkerhet kan datorsystems prestanda anges?
8. Hur kan den mänskliga delen (användardelen) i ett datorsystem urskiljas och analyseras?
9. Vilka prestandaanalysverktyg för datorsystem finns?
10. Hur kan prestanda belysas avseende
    - maskinvara
    - programvara
    - personvara?
11. Hur kan sammankopplingsbara (kompatibla) maskin- och programvarusystem utväljas från olika leverantörer?
12. Kan "maskinstorlekar" definieras?
13. Hur bör anbudsförhandlingar för datorsystem skötas?
14. Hur värdefulla är s k testkörningar i samband med utvärdering av datorsystem?
15. Hur används lämpligen simulering av datorsystem i samband med utvärdering?

16. Vilka typer av manuella analytiska belastningsberäkningar är lämpliga i samband med utvärdering?
17. Hur formuleras dator- och programvarukontrakt?
18. Hur utförs leveransprov för maskin- och programvaror?
19. Under vilka förutsättningar är köp av maskin- resp programvaror att föredra framför hyra?
20. Vem/vilka bör bedöma värdet av olika alternativa datorsystems egenskaper i den aktuella användarmiljön?
21. Under vilka förutsättningar kan värdet av olika egenskaper hos ett datorsystem vägas tillsammans?
22. Hur kan utfallet (resultatet) av ett installerat datorsystems arbete mätas?
23. Hur ofta bör resultatutfall kontrolleras?
24. Vilka manuella eller automatiska resurser bör tilldelas resultatmätning?
25. Hur bör resultatutfall bringas att återverka på informationssystemet, d v s hur bör uppföljning ske?

Tänkbar 'centraliserad' frågeställning för arbetet:

HYPOTES: Kapaciteten för ett datorsystem (maskinsystem plus användare) kan ej anges "in blanco", utan är beroende av den miljö inom vilket systemet avses användas.

alternativt

Det existerar inga allmängiltiga prestandamått för datorsystem.

Innehåll

1	Utgångspunkt för arbetet
2	Inledning
3	Utvärderingens innebörd
4	Målsättning
4.1	Olika typer av mål
4.2	Mängden ultimärmål
4.3.1	Uni-mål-situationen
4.3.2	Multi-mål-situationen
4.4	Målkonflikter i multimålsituationer
4.5	Intuition i beslutsfattande
4.6	Kvantifiering av mål
4.7	Konstruktion av målstrukturen
5	Databehandlingssystemegenskapernas natur
5.1	Egenskapernas kännetecken
5.2	Konkretisering av databehandlingssystemets
5.3	Nedbrytning av databehandlingssystemets egenskaper
6	Metodik för utvärdering
6.1	Beskrivningsprinciper
6.2	Utvärdering med absolut angiven måluppfyllelse
6.2.1	Kostnads/värde-teknik
6.2.2	Poängmetoder
6.2.2.1	Poängbegreppet och dess användning
6.2.2.2	Metoden med oviktad poängsättning
6.2.2.3	Metoden med viktad poängsättning
6.3	Utvärdering med relativt angiven måluppfyllelse
6.3.1	Metoden med oviktade parvisa jämförelser
6.3.2	Metoden med viktade parvisa jämförelser
6.3.3	Metoder med multipla jämförelser
6.4	Utvärdering med blandat absolut och relativt angiven måluppfyllelse
6.4.1	Begreppet urvalströsklar
6.4.2	Utvärderingsmetoden Electre
6.5	Utvärderingsarbetets resursförbrukning och utformning
6.6	Slutsatser ur redovisade utvärderingsmetoder

Innehåll, forts

7	Egenskaper för databehandlingssystem
7.1	Systemstorlekar
7.1.1	Grosch´s lag
7.1.2	Andra effektivitetsmått
7.1.3	Storlek och effektivitet
7.1.4	Effektivitet i användarmiljön
7.1.5	Undersökningar kring Grosch´s lag
7.1.6	Effektivitet i program- och personvaror
7.2	Egenskaper på lägre nivåer
7.2.1	Egenskapernas roll i offertarbetet
7.2.2	Beskrivning av typiska egenskaper

1 Utgångspunkt för arbetet

Föreliggande arbete har tillkommit som en följd av önskemål bl a från författarens sida rörande utökade kunskaper om utvärderingsprocessens innehåll och tillämpning, speciellt i samband med datorsystem. Att behovet härav är uttalat i vidare kretsar har speciellt under 1960-talets senare hälft framgått tydligt i samband med såväl statliga som industriella datasystemupphandlingar inom landet. Fr o m 70-talets ingång visar också möjliga anknytningar till europeiskt samarbete i datafrågor allt tydligare härpå, att döma av allt tätare publikationer från olika europeiska samarbetsorganisationer.

Konkreta faktorer, som haft betydelse för författarens beslut att söka bidra till kunskaper inom det angivna området, har varit flera svenska statliga datorupphandlingar under senare delen av 60-talet, och speciellt den rörande anskaffning av datasystem till Stockholms högre läroanstalter åren 1967-68. Denna affär inrymde ett flertal händelser som aktualiserade problem rörande beskrivning av datorsystems prestanda. Det framgick att för samtliga berörda parter entydiga prestandauppgifter för därvid främst aktuellt datorsystem icke syntes möjligt uppmätas eller på annat sätt anges. Då utgångspunkten därvid varit att det borde vara möjligt ange dylika "objektiva" prestanda (vilken hypotes, som det visade sig, alltså inte kunde verifieras) slutade denna affär i en viss otillfredsställelse för flera av de av affären berörda parterna.

För tillkomsten av föreliggande arbete har författaren uppburit stöd från Statens Naturvetenskapliga Forskningsråd

Successivt har därefter för författaren allt tydligare framvuxit en känsla av möjligheten att s k general purpose datorsystem[1] i många avseenden numera nått en sådan flexibilitetsgrad att à priori angivbara helhetsbelysande prestandauppgifter över huvud icke kan fixeras.

Det föreliggande arbetet är avsett belysa denna hypotes. Hypotesen kan sammanfattas till:

> Kapaciteten för ett generellt datorsystem är beroende av den miljö inom vilken systemet avses användas. Helhetsbelysande prestandauppgifter kan således endast anges i miljöberoende termer.

Vi kommer i arbetet löpande att återkomma till olika belysningar av denna hypotes, varvid dess relevans och verifierbarhet kommer att prövas.

---

1) 'General purpose' kommer i detta arbete att översättas med generellt. Som motsats kommer termen dedikerat datorsystem att användas. Begreppen syftar på systemens användning.

2 Inledning

De maskiner, som vill synas besitta den högsta utvecklingspotentialen av alla tekniska uppfinningar, är datorerna. I början av 1970-talet finns datorer som med underbar snabbhet löser synnerligen komplicerade problem. Vi har successivt kunnat notera hur datorsystemen utvecklas i en takt som vid jämförelse får det mesta att förblekna. Därmed har emellertid också styrproblem för utnyttjande av system i vilka datorer ingår kommit att bli centrala.

Utnyttjar vi våra datorer effektivt? Knappast, säger t ex D. Brandon i (1):

> "More than 40 % of all data processing installations in the U S are failures. Of those EDP installations that aren't failures, more than 90 % survive by crises. No industry has taken a tool of such unparalleled power and used it so badly".

Liknande formulering anges av H. Grosch (2):

> "... third generation computers within the U S are utilised at an efficiency level of about five percent".

(Att dessa herrar endast diskuterar situationen i Amerika har mindre relevans, läget i Europa är sannolikt knappast väsentligt annorlunda.)

Utsagor som dessa två senaste finner vi numera florera i mångfald bland "datorexpertis". Vad som därvid emellertid ofta glömmes bort är att omtala vad som avses med "effektivt utnyttjande" (motsv) av datorsystemen. Det finns skäl att betona att det i de flesta situationer ingalunda är önskvärt att se datorsystemen utnyttjas "maximalt", d v s upp mot 100 %. Koncentration bör i stället fästas på hur de samspelar med sin omgivning, d v s hur de hjälper människan, sett i ett vidare perspektiv.

Insikten att bl a datorernas verksamhet bör ses som en del i ett större system har relevans för uttalanden som t ex följande av Gyllenhammar (3):

> "Någonting som är mycket väsentligt och ganska skrämmande är att allting blivit så fruktansvärt komplicerat och att allting tar sådan tid .... Det som för 20 år sedan genomfördes på kort tid tar idag helt andra resurser och helt annan tid i anspråk. Vad vi än skall röra i när det gäller kommunens, stadens eller statens angelägenheter kräver insatser av en mångfald experter och olika funktioner".

Gyllenhammar syftar här på styrning av företag och andra organisationer. Överensstämmelsen med erfarenheter ifrån datormiljön är emellertid stor. Moderna datorers operativsystem, t ex, belägger i sig själva stora delar av tillgängliga datorresurser (minne, tid m m), och är ofta synnerligen komplicerade att styra. En tillämpningsrutin, som överflyttas från ett äldre till ett nyare datorsystem visar sig ofta kräva proportionellt mycket större resurser på det nya systemet. Även inom datorsystemen är alltså det mesta "mer komplicerat" nu än förr.

Allteftersom det system vidgas, i vilket en hjälpresurs utnyttjas, är det emellertid naturligt att denna resurs bör utnyttjas allt mindre "internt effektivt". Detta blir en följd av den höjda ambitionsnivå som följer av systemutvidgningen. Hänsyn måste tas till fler för resursen externa funktioner.

När vi höjer ambitionsnivån för våra system ökar vi deras svårstyrbarhet. Det förefaller finnas en tendens i vårt samhälle att öka systemens ambitionsnivå till ett läge något över vad som är lämpligt ur styr- eller hanterbarhetssynpunkt. Många av dagens automatiserade datorsystem satsar t ex på reelltidsbearbetning, trots att vi endast

relativt nyligen fått grepp om hur satsvisa system effektivt bör styras. En mänsklig parallell härtill finner vi formulerad i "Peters princip": 'En person befordras alltid upp till sin inkompetensnivå'. Kanske är detta karaktäristiskt även för många andra system.

Det talas idag på många håll om förverkligande av "totala informationssystem". Vad som därmed avses är icke fullt utrett. Uppenbart är att även i de mest "totala system mycket information måste flyta ostyrd, muntligen från person till person, utan dirigering från centralt håll. Den mänskliga naturen borgar härför. Snarare än att tala om "totala" system borde därför lämpliga delar av system betraktas. En nivå borde i varje enskilt fall definieras, vid vilken gränsen bör gå mellan styrd och ostyrd informationsförmedling. Och denna gräns kan i vissa fall vara oskarp, för att medge flexibilitet i det avsedda systemet.

Samspelet mellan människa och system är ännu endast utrett i begränsad omfattning. Hur reagerar människan då hon på tjänsterummet/verkstaden får snabb direkt tillgång till relevant information? Kommer terminalerna delvis att användas som leksaker? Vilken samspelseffektivitet uppnås? Hur långt vill människan låta sig 'programmeras'?

Det kan vara motiverat att komplettera begreppen programvara och maskinvara med personvara, varmed avses faciliteter för kommunikation mellan människa och program. Vi finner i föreliggande arbete att vi vid datorsystemkonstruktion i tur och ordning bör beakta personvaran, programvaran och maskinvaran. Vi kan åskådliggöra detta samspel mellan människa och maskin i följande figur:

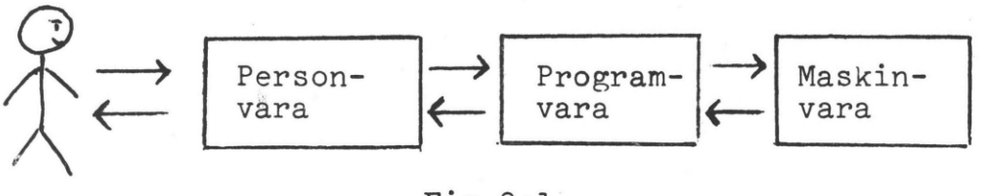


Fig 2:1

Till vidare belysning av denna uppdelning kommer vi senare att återkomma.

I mitten av 1960-talet ansågs ett normalt datorsystems investering i programvara och maskinvara ungefärligt böra fördela sig som 50-50. Det har på många håll, bl a av Alt i (4), indikerats att vid (början eller) mitten av 1970-talet programvaran skulle kräva upp mot 80 % av investeringarna. Det blir då uppenbart att, enligt (4), "even a drastic reduction in the cost of hardware .... can have only a minor effect on the cost of computing as a whole". Sålunda finner vi koncentration på personvara och programvara naturlig i framtida systemplanering.

Det är förvånande att, enligt t ex beskrivningar i fackpressen (ex Laska i (5)), i många hittills förverkligade system planeringen för arbetet med implementeringen lämnat så mycket övrigt att önska. Langefors anger i (6): "People tend to underestimate the complexity of an imperceivable system". Många system uppfyller fordringarna för att kunna anses som 'imperceivable'. Det talas ofta om att datorsystem visat sig dubbelt så kostnadskrävande som planerat. Denna faktor två är dock med säkerhet approximativ. Ett av skälen till dylika undervärderingar av erforderligt implementeringsarbete är säkert frånvaron av lämpliga metoder för problembeskrivning. Så t ex anger Alt i (4), syftande på "high failure rate": "I submit that the real root of this evil is the inadequacy of our ordinary human language for describing the problem before programming begins: specifying all possible combinations of circumstances which can arise in the course of the problem, and stipulating how the program is to react in each case". Alts kommentar syftar främst på språkliga brister, det kan finnas skäl att, som t ex Bubenko i (7), utvidga detta till att även innefatta brister avseende principiell beskrivningsmetodik.

Alt konstaterar även att "we should now add that there are many problems within the capacity of computers which are too complicated for humans even to describe .....".

Det är ur praktiska erfarenheter tyvärr uppenbart i vilken alltför begränsad utsträckning systematiska implementeringsprinciper kommit till användning i realiserade system. Det bör dock tilläggas att på tillräcklig detaljnivå praktiska och teoretiskt hållbara systemeringsprinciper ännu inte (1971) utvecklats fullständigt. Langefors anger i (8): "Det har naturligtvis .... speciellt intresse att studera sådana faser av systemarbete för data- och informationssystem, för vilka tidigare ej mycket erfarenhet samlats eller metoder ej utvecklats. Detta gäller i första hand för analys, utformning och utrustningsanpassning ....".

Vi kan konstatera att det på många håll praktiskt visat sig råda ett från början icke önskvärt förhållande mellan avsedda och verkligt uppmatta systemresursbehov. Även jordnära faktorer har medverkat härtill. Det har t ex blivit bekant att under 60-talet anskaffning av flera datorsystem i själva verket överenskommits på de av verkställande direktörer frekventerade golfbanorna, helt utan tillfredsställande bakomliggande analys och utvärderingsarbete. (Detta låter sig dock sannolikt svårligen empiriskt bevisas).

Det kan uttryckas förhoppningen att föreliggande arbete skall kunna kasta ljus över vissa av problemen kring utvärdering av databehandlingssystem, och att golfbanekontrakten därmed skall komma att minska i antal.

3 Utvärderingens innebörd

"Att jämföra datorsystem är
som att jämföra människor,
d v s omöjligt"

Utvärdering är en process som vi människor ofta är sysselsatta med. Vad innebär begreppet? Utvärderingen innebär att vi jämför ett visst systems egenskaper med en uppsättning önskvärda egenskaper, vilka senare vi hoppas systemet skall besitta i hög utsträckning. Ofta har begreppet utvärdering emellertid kommit att expanderas till att "ur en mängd tänkbara system utvälja ett som i önskad utsträckning besitter erforderliga egenskaper". För att kunna fatta beslut om val av "bästa" system, fordras dock därvid att vi först applicerar metoder för att kunna få grepp om egenskaperna för samtliga de system som kandiderar till att bli utvalda.

Låt oss belysa utvärderingsbegreppet med ett synnerligen enkelt exempel. Betrakta en husmor (eller husfar, för den delen) som i en affär har att besluta sig för inköp av ett halvt kilo smör eller ett halvt kilo margarin. Vilketdera skall hon (vi utelämnar här 'han' av bekvämlighetsskäl) väljas? Låt oss utgå ifrån att situationen är en verklig valsituation för henne, och att hon alltså inte bestämt sig på förhand. Hennes val är uppenbarligen beroende på vad hon relaterar beslutet till. Är hennes enda föresats att hålla kostnaderna nere, så väljer hon margarinet, eftersom detta är billigare. Men troligen kan andra föresatser vara aktuella samtidigt. Hon anser kanske att smör är godare, medan hon dessutom är medveten om smörets innehåll av animaliskt fett, vilket sannolikt i det långa loppet befrämjar åderförkalkning. För margarinet talar alltså

- det lägre priset
- avsaknaden av animaliskt fett,

medan smöret kan tillgodoräkna sig

- den godare smaken.

(Vi lämnar här övriga tänkbara faktorer därhän.) För att bestämma sig tvingas hon nu att på något sätt evaluera eller värdera de tre egenskaperna mot, sätta dem i relation till, sin (och kanske familjens) egna önskningar. Inte bara måste hon hålla antalet aktuella egenskaper i minnet, hon måste också beakta vilken betydelse egenskaperna har för henne. Hennes val kommer att bli beroende av vilken vikt hon tillmäter de olika egenskaperna. Är smörets godare smak viktigare för henne än margarinets lägre pris och bättre hälsoeffekt? I de flesta valsituationer, liksom i denna, kan egenskaperna inte mätas i samma enhet. Hon kan då söka omvandla egenskaperna till någon lämplig gemensam referensskala, bedöma hur de erhållna resulterande mätvärdena kan sammansättas, utföra denna sammansättning, och därefter fatta beslutet.

Det är inte alltid möjligt för henne att finna denna gemensamma referensskala. I vårt exempel bör detta framgå. Med vilken gemensam enhet skulle smak, hälsoeffekt och pris kunna mätas? I sådana fall blir hennes beslut avhängigt hennes intuition. Vi skulle kunna kalla detta en icke-formell beslutssituation.

I andra situationer kan det betraktade systemets egenskaper verkligen relateras till samma skala, en formell beslutssituation. Egenskaperna kan då ges mätvärden, som beskriver hur väl de uppfyller önskningarna, och dessa mätvärden kan i gynnsamma fall adderas till varandra. Därvid erhålles ett slutligt mått på systemets 'värde' i den betraktade situationen.

Vi bör påminna oss att vi mycket ofta ställs inför situationer liknande den i exemplet nämnda. Detta gäller inte bara rörande vår konsumtion, utan i samband med de flesta av våra dagliga beslut. Det är sannolikt att vi därvid, medvetet eller (oftare) omedvetet, mentalt utför processer liknande de ovannämnda. Det är emellertid möjligt att det

må brista något i stringensen vid snabba och frekventa beslut. Så t ex är det tänkbart att vi ofta intuitivt sammansätter värdeomdömen, som egentligen relaterar sig till olika önskemål, och där sammansättning i stället borde ge ökad plats för prioritering.

Utvärdering av datorsystem skiljer sig inte principiellt från utvärdering av andra typer av system. Svårigheter i samband med utvärdering kan i stället betraktas som generella, och giltiga för vad vi än applicerar dem på. En viss gradskillnad kan vi dock notera. Ju flera separata mål som är giltiga i den betraktade miljön, desto svårare blir det ofta att finna det 'lämpligaste' (eller minst olämpliga) av ett antal alternativa system. Utvärderingen är mer krävande i en multimålsituation, därför att i det fallet systemets egenskaper måste relateras till flera skilda mål. Speciellt besvärlig blir situationen då dessa mål delvis är motstridiga.

Vi kan också betrakta utvärderingen ur de för utvärdering aktuella systemens synvinkel, och oberoende av målstrukturens utseende. Ju flera skilda egenskaper systemen besitter, desto flera faktorer måste vi i utvärderingsarbetet ta hänsyn till, och desto mer utvärderingsresurser krävs. Vi kan följaktligen konstatera att

> ju flexiblare och mer omfattande system som skall utvärderas, desto mer resurser krävs för utvärdering av desamma.

Då bl a flexibilitet och systemomfattning är nära anknutna till begreppet ambitionsnivå, kan vi även säga att

> ju högre ambitionsnivå för de betraktade systemen, desto mer resurskrävande utvärderingsarbete.

De betraktade systemens egenskaper kan naturligtvis vara kända med större eller mindre visshet. Uppenbart är att vi knappast kan göra anspråk på att ha utfört en utvärdering tillfredsställande, om vi inte tillräckligt noggrant informerat oss om de betraktade systemens egenskaper. En kedja är aldrig starkare än sin svagaste länk, och ett aldrig så raffinerat utvärderingsarbete, som bygger på dålig systemkännedom, innebär därför ett visst slöseri med resurser. Mängden resurser som tilldelas själva utvärderingsarbetet bör därför stå i lämplig relation till detaljkännedomen om de betraktade systemen. Har vi ingen möjlighet att, på tillgänglig tid, få tillräcklig information om systemegenskaperna, lönar sig därför knappast ett alltför omfattande utvärderingsarbete.

Våra möjligheter att få grepp om de för utvärdering aktuella systemens egenskaper är i en dominerande utsträckning beroende av mänskliga värderingar. Vi kan visserligen tänka oss system, vilkas egenskaper kunde greppas "objektivt", men i alla praktiska situationer blir dylik "objektivitet" endast en sammanfattning av ett antal mänskliga subjektiviteter. Hur 'maskinellt' (el motsv) vi än mäter systemegenskaper, kvarstår det faktum att de mätdon vi förlitar oss på, konstruerats av människor, och "programmerats" (eller instruerats) att mäta enligt vissa givna regler. Konstruktörernas subjektiviteter framträder därvid. Det är alltså rimligt att från början konstatera att subjektiva mänskliga värderingar blir nödvändiga i utvärderingsarbetet. Subjektiviteterna kan emellertid vara mer eller mindre väl dolda. I en klokt upplagd utvärdering redovisas dessa subjektiviteter öppet, t ex genom namn på aktivt aktuella personer, och söks ej dölja de för systemvalet ansvariga instanserna. Vi konstaterar alltså att

> utvärdering innebär att vi överlåter beslutanderätten om systemval åt ett antal personer, och ej åt någon formaliserad objektivitet.

Vi kommer att återkomma till synpunkter på utvärderingsgruppens lämpliga utseende.

Hittills har vi anlänt till slutsatsen att utvärderingen innebär fattande av på mänskliga bedömningar baserat beslut om vilket av ett antal (två eller flera) skilda system som kan anses ha egenskaper som bäst är anpassade (eller anpassningsbara) till målstrukturen för en given miljö.

Tillämpat på situationer rörande datorsystem är det betydelsefullt att framhålla att anpassningen av ett dylikt system till en mänsklig miljö, exempelvis ett företag, koncentrerar uppmärksamheten till den samlade effekten av användare och person/program/maskinvara. Utvärderingen innefattar därvid att arbeta fram beslutsinformation rörande om dator(er) över huvud taget skall anskaffas, eller förnyas. Det kan motiveras med införande av ett datorsystem inom ett företags informationssystem endast om det kan visas (eller åtminstone göras tillräckligt troligt) att detta, totalt sett, ökar företagets måluppfyllelse. Ställningstagande rörande datoranskaffning kan sålunda knappast ses som ett i företaget isolerat och rent datatekniskt beslut.

För att nu närmare kunna studera utvärderingsprocessen behöver vi

1 analysera aktuella målstrukturer, d v s mål till vilkas uppfyllelse datorsystemets existens förväntas kunna bidraga

2 analysera aktuella datorsystems egenskaper, såväl på en allmän som på en så långt möjligt detaljerad nivå

3 formalisera metoder för beslutsfattande rörande val av system med egenskaper enligt 2 ovan med bästa måluppfyllelse enligt 1 ovan.

Vi kan sammanfatta dessa tre punkter i följande figur:

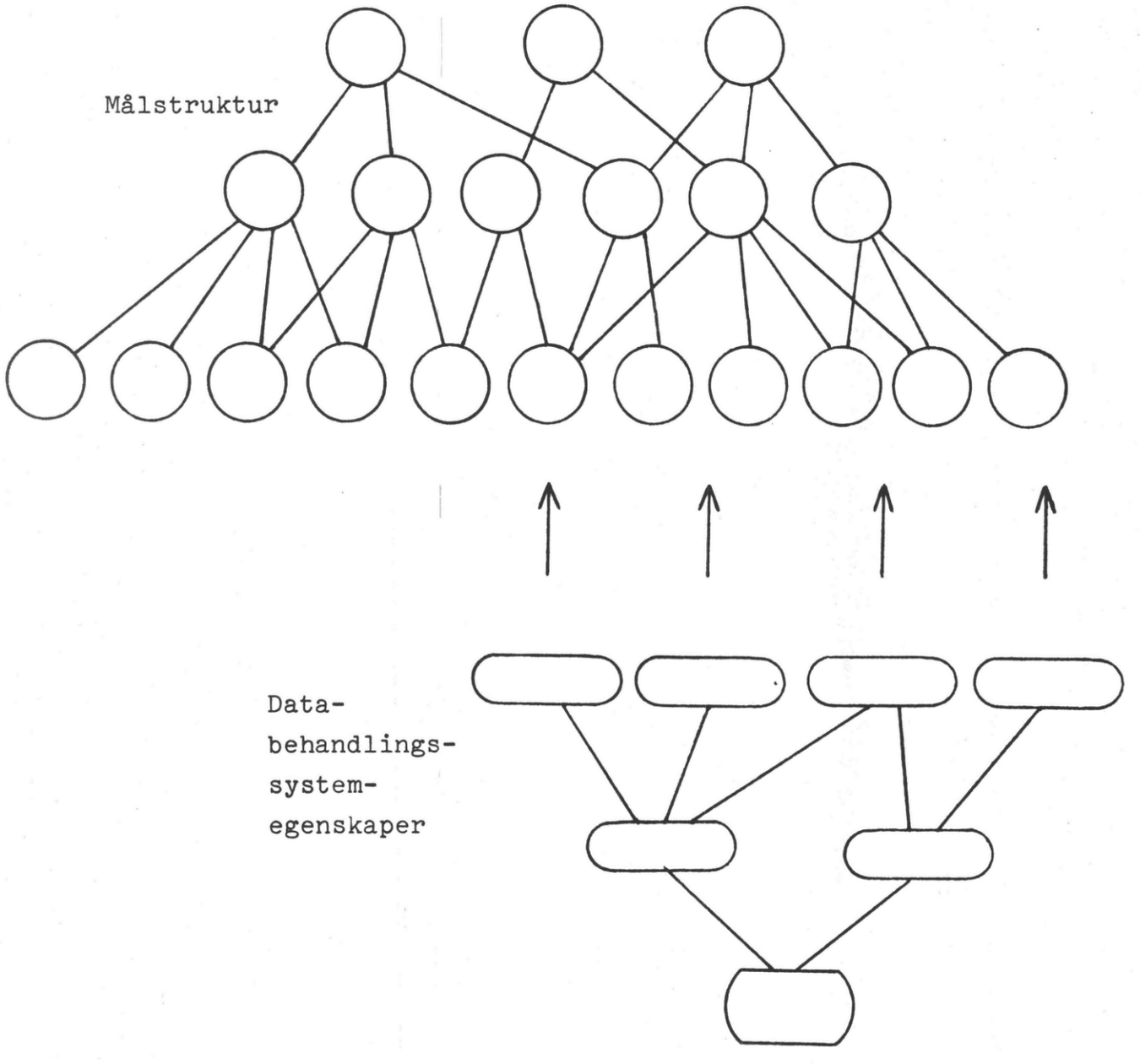


Fig 3:1 Schematisk bild av systemutvärdering

4 Målsättning

4.1 Olika typer av mål

För oss individer såväl som för företag och andra samhällsintressenter är det uppenbart att olika typer av mål existerar. Som exempel kan vi betrakta

1 maximal lycka

2 tillfredsställande ekonomi.

Det vill synas klart att vi knappast kan betrakta dessa mål som "likvärda". I något eller några avseenden skiljer de sig från varandra. När vi i fortsättningen kommer att tala om mål på olika nivåer bör vi hålla i minnet att generella utsagor om samband mellan explicita mål inte kan göras. För många individer må gälla att målet 2 ovan kan ses som "underställt" målet 1. Vi kan säga att målet 2 må vara en precedent till målet 1, och vi menar då att ett bidrag till målet 2 också automatiskt är ett bidrag (dock kanske ej lika stort) till målet 1 (medan motsatsen ej gäller generellt). - Det kan emellertid mycket väl finnas personer, för vilka det motsatta sambandsförhållandet gäller. - Vi kan konstatera att olika typer av mål existerar, men att dessa för olika berörda må ha skilda utseenden. Allmänt sett kan precedensförhållanden existera mellan olika typer av mål.

Det är alltså naturligt att betrakta ett företags olika mål som ett system. Vi måste analysera målsystemet för att kunna angripa, och helst mäta, hur ett datorsystem kan bidra till måluppfyllelsen.

Vi kommer att använda termen målstruktur för den samlade mängden av företagets mål samt de samband som må gälla dem emellan. För att egenskaperna för datorsystemet skall kunna relateras till målstrukturen blir det naturligt att bryta ner målstrukturen i mål (delmål) på lämpligt

låg nivå. Vi kan t ex knappast direkt avgöra huruvida ett snabbt magnetbandssystem eller ett långsamt skivminnessystem bäst medverkar till att inom företaget kunna hålla en god arbetsmiljö (under förutsättning att "god arbetsmiljö" är ett gott mål för företaget). - Vårt resonemang kan belysas genom följande citat ur Langefors (9), sid 122:

> "Målsättning - rätt utförd - definierar vad som är det väsentliga "totalresultatet" hos varje delsystem och, genom att ge "operationell" ("mätbar") form åt delmålen, gör det möjligt att följa upp väsentligheter (ur totalsystemets synpunkt) men lämna valet av alternativa vägar att nå det väsentliga åt resp delsystem själva".

I de resonemang om mål på olika nivåer, som vi för i detta avsnitt, kommer vi att ansluta oss till begrepp och benämningar, givna i Langefors (9).

Sålunda används följande begrepp:

Ultimärmål (eller intention): de yttersta syftena - anledningen till att verksamheter, eller systemet hålles i drift.

Önskemål det vi önskar, eller förhoppas, är möjligt.

Operativt mål mål som tages som medveten bakgrund för styrning eller planering av operativ verksamhet, d v s av de operationer som avses bli utförda under en viss tidsperiod.

Operationellt mål: mål, där det kan avgöras huruvida det uppfyllts, ett mätbart mål.

Primärmål: mål som måste tilldelas högsta vikt vid beslut där olika mål kommer i konflikt.

Ofta kan ej enkelt avgöras i vilken kategori enligt ovan ett visst mål bör anses höra hemma. Generellt kan härom inga utsagor göras, då självfallet endast företaget självt kan avgöra aktuella måls karaktär. Vi ger dock här en

exemplifierande lista på olika företagsmål som noterats i en undersökning av Johsen (ref (10), sid 56). Där diskuteras bl a huruvida företag kan anses sträva mot ett eller flera mål (objectives). Följande mål har angivits av intervjuade företag:

"One-objective formulations

- Provide a given sort of service
- A certain measure of profit
- At least K per cent on investment
- Profit maximation in the long run

Two-objective formulations

- Growth and profit
- Growth and independence
- Production and sales of certain products
- Transform raw materials into consumers goods, and financial security
- Profit and service
- Profit and price influence
- Profit and live in peace with surroundings
- Expand service X and service Y

More-than-two objective formulations

- Market expansion, profit, service, benefit to the external society
- Survival, minimum ("at least") profit, customer service
- Make money, expand, consolidate."

Vid studium av ovanstående lista framgår att nämnda mål i flera fall förefaller befinna sig på olika "nivåer". De flesta av målen kan knappast betecknas som ultimärmål, flertalet är i stället av närmast operativ karaktär. Det kan dock vara rimligt att anta att för många företag

t ex målet "överlevande" bör ses som ett önskemål, däremot knappast som ett ultimärmål; överlevande är knappast ett yttersta syfte för verksamheten, dess uppfyllande kan dock bidra till uppfyllande av företagets ultimärmål. Vi kan ovan finna tydliga exempel på mål av operativ karaktär, t ex minimum ("at least") profit.

Det är tydligt att samband kan tänkas existera mellan mål av ovan nämnda typer, genom att ett bidrag till uppfyllande av ett visst mål direkt innebär även ett bidrag till uppfyllande av ett annat, "högre" mål. För att kunna klarlägga dylika samband, eller precedensförhållanden, är uppenbarligen de ovan angivna begreppsdefinitionerna värdefulla.

Det förda resonemanget leder oss fram till en tänkbar nivåuppdelning av olika företagsmål. Denna kan ges en form som i nedanstående figur (en smärre modifiering av angivelser i Langefors (11)):

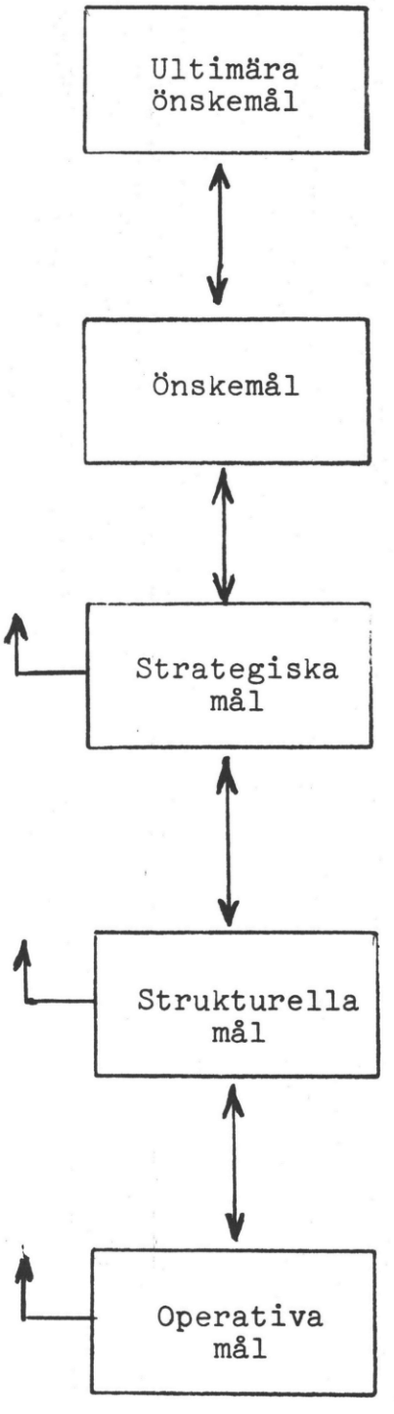


Fig 4.1:1 Mål på olika nivåer

Vi kommer i detta sammanhang ej att närmare gå in på den målnivåuppdelning som ges i fig 4.1:1. Vi konstaterar blott att ju högre upp i målhierarkin som ett mål befinner sig, desto mera okänsligt kan det anses vara för påverkan från den konkreta omgivning, den miljö, där det befinner sig. Som exempel finner vi det naturligt att ett mål av ultimär karaktär för många företag: "vinstmaximering på lång sikt" ej bör behöva påverkas (omformuleras e dyl) av mera kortsiktiga vinstutfall eller andra effekter. Vi kan formulera detta som att mål på högre nivåer bör vara mera resistenta mot miljöpåverkan än mål på lägre nivåer.

Ytterligare ett klargörande vad gäller olika typer av mål krävs för våra fortsatta diskussioner. Vi har nämnt det synnerligen viktiga begreppet operationalitet, d v s mätbarhet, hos mål. Vi behöver emellertid även diskutera hur väl ett operationellt mål uppfylls. Därvid är det naturligt att skilja mellan mål som ej avses resp som avses explicit uppnås. Vi kan uttrycka detta på det viset att det må kunna vara önskvärt med obegränsad strävan i en viss målriktning, men att ett uppnående av en viss uppfyllelsenivå ej är direkt intressant.

Vi kommer att använda benämningarna satisfierande för mål vars uppfyllande innebär uppnående av en viss nivå, en tröskel el liknande, medan vi kallar ett mål optimerande om vi avser det önskvärt med en "uppåt" icke begränsad strävan i en viss riktning. Ett exempel på ett satisfierande mål må vara: "uppnående av p % avkastning på investerat kapital", medan "maximal lycka för alla anställda" kan anses som ett optimerande mål. Vi kommer att finna att skillnaden mellan optimerande och satisfierande mål är av avgörande betydelse vid diskussion om måluppfyllelse. Som kommentar rörande exempel på satisfiering citerar vi här W.J. Karplus, som i (12), sid 76, rörande val mellan satsvis och direkt bearbetning, anger:

> "In many cases, the user objectives and requirements for the system are such that the one-line character

> of the system is, for all practical purposes, mandatory. There is no other way of meeting the requirements. Examples of such systems are travel reservation systems, command and control systems, and stock quotation systems. These systems have the common requirement that the user have ready access to the data in the machine; the answers would be of little use if he had to wait the minutes or hours required of batch processing. Therefore, in these cases, the question is not whether batch processing is better. Rather, the question immediately becomes one of the best design of the one line system".

Här kan tilläggas att Karplus med "little use" sannolikt avser (bör avse) att informationens värde (med för lång väntetid) understiger en satisfierande tröskel.

Den kategorisering av mål, som ovan givits, rör målens typ eller art. Vi bör dock hålla i minnet att ett visst mål icke á priori kan ses som t ex ett ultimärmål. Mål som är operativa för ett visst företag kan mycket väl tänkas anses som ultimära för ett annat. Endast i en given miljö, exempelvis för ett bestämt företag eller en given grupp människor, kan fixeras karaktären för ett visst mål. Kategorisering av mål till olika typer är alltså miljöberoende. T o m operationalitet hos ett mål är beroende av dess omgivning, i detta fall hur mätskalor och mätinstrument är beskaffade i det aktuella fallet.

Emellertid bör vi notera att inom samhället olika grupper av företag eller andra organisationer existerar, för vilka miljöerna är likartade. Inom dylika grupper kan självfallet samma mål vara av likartad karaktär.

Om miljödefinitionerna kunde skärpas, d v s tillräckligt exakta gruppgränser kunde fixeras, skulle basen för gemensamma diskussioner om måluppfyllelse inom grupperna stärkas.

Ett exempel på måls miljöberoende finner vi i Bubenko-Ohlin (13), sid 303, där följande figur avser visa hur aktiva funktioner hos ett operativsystem på

olika sätt kan bidra till uppfyllese av olika intressenters mål:

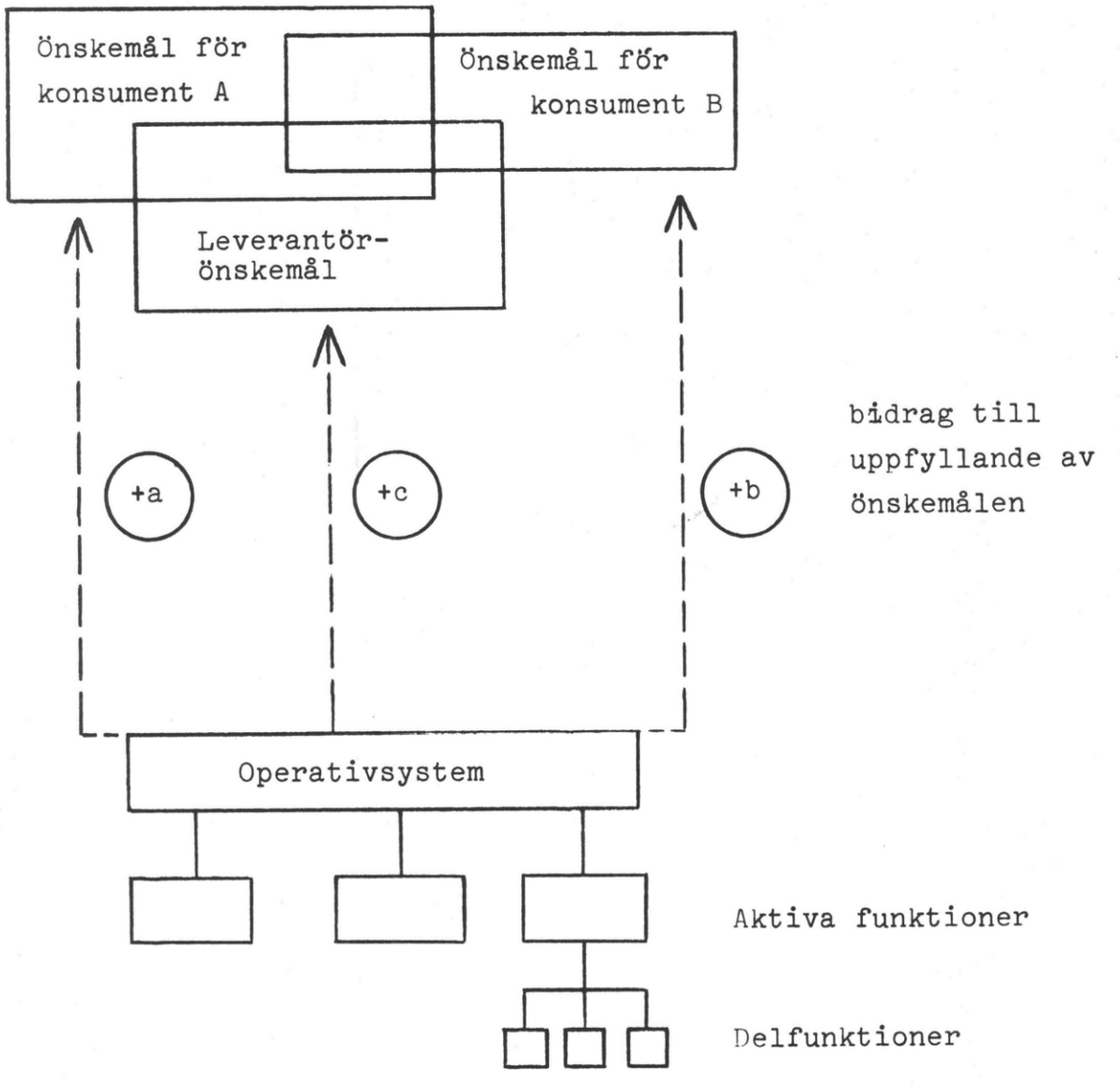


Fig 4.1:2 Måls miljöberoende

4.2 Målsättningens dynamik

Som all verksamhet i företag är även målsättningen exponerad för förändring, orsakad av påverkan. Vi diskuterar i detta avsnitt hur olika former av påverkan kan vara av betydelse för ett företag som överväger införande av ett (nytt) datorsystem i syfte att förbättra företagets måluppfyllelse.

Beträffande förändring av företagsinriktning konstaterar vi endast här att det är rimligt förmoda att ändring av höga mål inträffar med låg frekvens. Dylik förändring kräver omstrukturering av såväl lägre mål som (sannolikt även) den operativa verksamheten, och bör därför knappast beslutas ofta. På lägre nivåer i målhierarkin kan dock företaget mera frekvent besluta sig för ändringar. - Den förflutna tiden från det en viss målstruktur beslutats kan vara en direkt påverkande faktor för förändring. Mål kan ofta ha ett begränsat tidsmässigt definitionsområde, eller på annat sätt vara definierade som en funktion av tiden. - Inträffade externa händelser av många slag är uppenbarligen möjliga orsaker till målförändring. Detta kan röra sig om konkurrenters agerande på marknaden, politiska beslut i samhället (statliga ränteändringar, skatter, socialpolitik, och mycket annat) m m som visar sig kräva en förändring av företagets målstruktur.

Förändring av målinriktning med anknytning till ovanstående resonemang diskuteras bl a av J. Martin och A.R.D Norman i (14), sid 364:

> "Under conditions of external threat most members of human organications cooperate and concentrate on the achievement of primary objectives".

Vi kan tolka detta förhållande på två sätt: Antingen är målstrukturen att betrakta som statisk, och vi koncentrerar oss (under påverkan av t ex externt hot) på en viss del av målstrukturen, d v s vi förändrar vår önskan om måluppfyllelse dynamiskt, eller vi påverkar direkt målen

för verksamheten dynamiskt, och betraktar oss själva som mera statiska. Vilken tolkning som väljs är ett personligt val. För alla dessa typer av målförändringar (och andra) kan vi konstatera det uppenbara att tidsrymden för att anpassa företaget till den nya målstrukturen måste vara kortare än den tidsrymd för vilken de nya målen är giltiga. Hinner ej denna anpassning av företaget ske i tid, kommer företaget att kunna försättas i "svängningar", d v s instabilt tillstånd. Vid all målsättning bör därför noga klarläggas de ansatta målens giltighetstid.

En vanlig orsak till förändring av mål för företaget är uppmätande av utfall av företagets verksamhet. Det befinns att måluppfyllelsen under antagna resursförutsättningar inte visar sig acceptabel, och en ändring av såväl målstruktur som allokerade resurser kan beslutas.

Ett företag som i ett nytt informationssystem inkluderar datorutrustning finner ofta att just existensen av dylik utrustning blir en orsak till målförändring. Det rör sig här vanligen om modifiering av mål på lägre nivåer. Vi kan säga att nya mål induceras. - Det har ofta konstaterats att "aptiten" successivt växer när modern datautrustning bringas tillhanda. Efter hand som applikationsmiljön successivt formaliseras och på andra sätt anpassas till det nya informationssystemet (och tvärtom) befinns att nya, ännu ej påtänkta, databearbetningar blir aktuella.

Vi kan åskådliggöra detta förhållande, alltså informationssystemets påverkan på företagets målstruktur, i en figur:

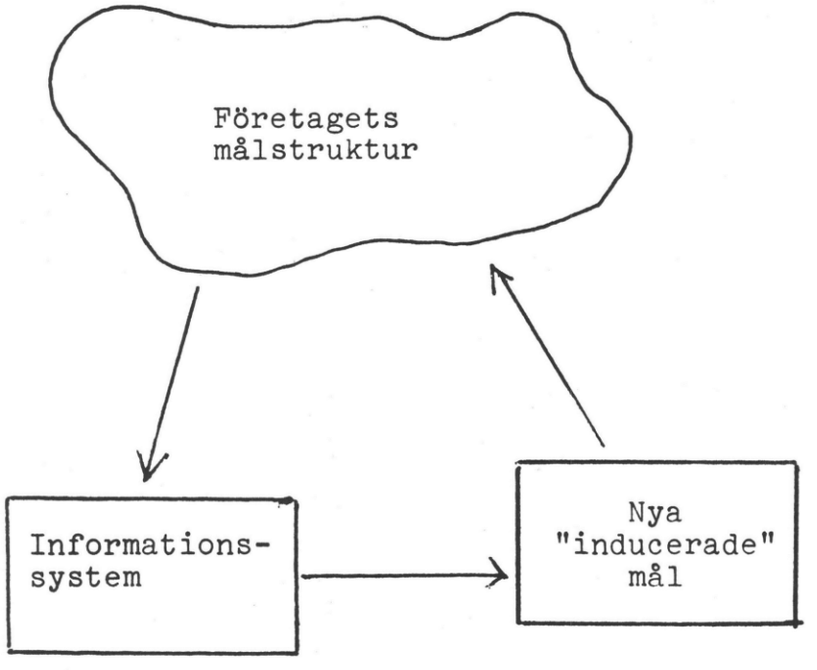


Fig 4.2:1 Måldynamik

Vi kan konstatera att vi i praktiken sålunda ofta har att göra med ett iterativt dynamiskt målsättningsproblem. För att kunna behandla detta måste vi fixera en tidpunkt, vid vilken målanalysen genomföres, men fästa vikt vid att vår målstruktur utformas på ett flexibelt sätt, så att inte alltför mycket arbete spolieras av en senare inträffad förändring.

Som en enkel illustration kan vi i ett exempel anta att förändringen utgörs av ökade krav på volymen bearbetade data enbart. Vi inför även i exemplet andra starkt förenklade förutsättningar för utvecklingen i tiden. Antag att företagets krav på bearbetning stiger linjärt med 25 - 30 % per år (ett ej ovanligt förhållande), samt att datorkapaciteten p g a förbättrade programvaror m m stiger linjärt med 10 % per år. Vi utelämnar i åskådlighetssyfte allt annat tidsberoende än det ovan givna. Hur skall vi dimensionera datorutrustningen om vi antar en tidslängd för densamma på 5 år?

En figur kan åskådliggöra förloppet:

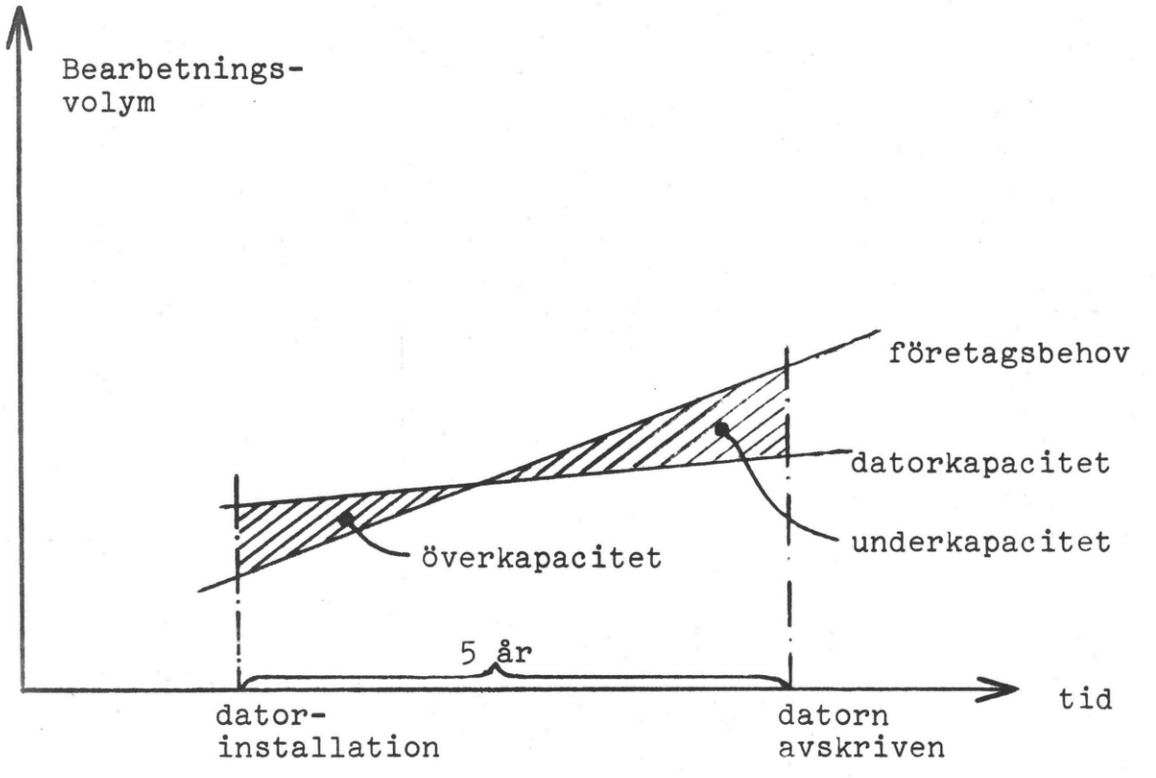


Fig 4.2:2

Företaget eftersträvar att i inledningsskedet ha så liten överskottskapacitet som möjligt samt att i avslutningsskedet ha så liten underskottskapacitet som möjligt.

I fig 4.2:2 utgörs de streckade ytorna av överskotts- resp underskottskapacitet. Summan av dessa ytor bör alltså minimeras. Symmetrin i problemet leder till att överskottskapaciteten (såväl som underskottskapaciteten) bör vara noll vid tidsintervallets mittpunkt. Med de givna förutsättningarna (25 - 30 % ökning på 5 år = 2.45 - 2.85, säg 2.65, samt 10 % ökning på 5 år = 1.45) finner vi geometriskt (eller på annat sätt) att den erforderliga överkapaciteten vid installation bör vara ca 90 %, d v s datorkapaciteten då bör vara 1.9 gånger det initiala behovet.

Detta räkneexempel är uppenbarligen förenklat nära nog till overklighet, (ex hur mätes dylik databearbetningskapacitet?) men kan dock tjäna som grov belysning till hur eventuellt ökade behov bör kunna tas hänsyn till på ett tidigt stadium av ett systems tillgänglighet.

Dynamiken i målsättning för företagsstyrning har behandlats av ett flertal författare. Langefors påpekar i (9), sid 15:

> "Det är viktigt att inse att i den dynamiska miljö där företagen av idag verkar ständigt nya problem uppstår. Man kan därför ej räkna med att hinna uppställa mål och formell metodik som klarar dem alla. Följden blir att jämsides med ett formaliserat informationsbeslutssystem, som kanske också automatiserats med datamaskiner, måste finnas ett fungerande informellt system. Kommunikationen mellan de båda systemen måste ges en god utformning."

En belysning av förändringar som ofta uppträder i samband med installation av ett datorautomatiserat informationssystem ges av Kanter i (15), sid 243 ff. Han anger där

där bl a fyra rubriker som speglar den nämnda dynaiken:

1 "Advanced applications". Här diskuteras företagets ständiga otålighet med existerande förfaranden, och de ökande servicekrav som ofta introduceras av det nya informationssystemet.

2 "Systematic, logical growth". Problem kring förändringar baserade på normal expansion av det framgångsrika företaget behandlas.

3 "Reduced systems implementation cycle". Här berörs konkreta följder av det "vidgande" gapet mellan implementeringsplaneringen och den verkligt utförda implementeringen.

4 "Economical, efficient operation". Här tas de konkreta problemen kring ökade krav på datorkapacitet upp. Därefter nämns avslutande: "The planning process should be based on a careful cost/benefit analysis, the background of the company's over-all goals and objectives, and the information systems that are required to reach them".

Som sammanfattning kan vi konstatera att dynamiken i målsättningen kräver en viss flexibilitet hos det styrda systemet, d v s företaget. Alltför mycket resurser bör dock ej investeras i åstadkommandet av denna flexibilitet. Ett 'exekutivt optimum' bör eftersträvas. Langefors formulerar i (16), sid 4:

> "Some reasonable balance will obviously have to be obtained between the efforts put into goal-setting, on one hand, and the decision analysis to achieve the goals, on the other hand".

Emedan omgivningens förändringar sällan låter sig förutsägas i detalj, må det emellertid ofta visa sig svårt att praktiskt finna detta exekutiva optimum. Man tvingas måhända nöja sig med att successivt komma en bit på vägen.

4.3 Mängden ultimärmål

Vi har ovan konstaterat att det finns skäl att betrakta företagets samlade mängd av mål som ett system, som för att kunna greppas i syfte att nå god måluppfyllelse, bör analyseras på liknande sätt som företaget självt. Vi har därvid bl a funnit lämpligt diskutera mål på olika nivåer. Vi har dock inte närmare diskuterat det explicita utseendet av detta målsystem, eller denna målstruktur.

Målstrukturen för det enskilda företaget må ha ett godtyckligt utseende. Vissa egenskaper hos målstrukturer i allmänhet har emellertid betydelse för hur måluppfyllelsen bör dirigeras. Vi diskuterar här först utseende av mängden av de "övre" målen. Vi kommer att finna att olika angreppssätt för uppfyllelseanalysen blir aktuella om denna mängd består av ett eller flera element. (Vi bortser här från fallet att inget ultimärmål existerar, vilken situation sannolikt skulle kräva omdefinition i det enskilda fallet.)

4.3.1 Uni-mål-situationen

I fallet att företaget som ultimärmål har ett enda mål är situationen förhållandevis gripbar. Är målet att betrakta som satisfierande, eftersträvas naturligen endast uppnående av den givna satisfieringsnivån. Någon optimering eller förbättring därutöver är då icke aktuell. Som exempel på en dylik situation kan vi tänka oss ett företag vilket som enda ultimärmål strävar mot "uppnående av p % avkastning på investerat kapital". Företaget kommer icke att anse sig behöva förbättra måluppfyllelsen över p %. (Däremot kan, då p % uppnåtts, möjligen diskussion resas om lämpligheten av att höja denna nivå. Därmed ändras emellertid målstituationen, och en ny satisfierande nivå fastslås, och eftersträvas.)

För fallet att företagets enda ultimärmål är av optimerande typ kan situationen tänkas vara den att en "nyttofunktion" (el motsvarande) definieras, och största tänkbara värde för denna funktion eftersträvas. Vi kan då aldrig tala om att "ha uppnått" målet, eftersom ständiga förbättringar av måluppfyllelsen kan tänkas, oberoende av utgångsnivå. "Ett optimerande mål kan aldrig uppnås" är en formulering som vi finner naturlig. Som exempel på denna situation kan vi tänka oss ett företag där enda ultimärmål utgörs av "största tänkbara arbetstillfredsställelse för de anställda".

I uni-mål-situationen kan givetvis icke konflikter på ultimärmålnivå bli aktuella. (Dock kan dylika uppstå på lägre nivåer.)

4.3.2 Multi-mål-situationen

Vi har tidigare noterat exempel (Johnsen i (10)) på angivelser där företag anser sig sträva mot fler än ett ultimärmål "samtidigt". Denna situation reser möjligheter till konflikter, som kan vara svåra, i vissa fall omöjliga, att lösa upp. Låt oss betrakta tänkbara situationer som kan vara aktuella med avseende på satisfierande resp optimerande karaktär hos ultimärmålen:

Samtliga ultimärmål är av satisfierande typ.

I fallet att inga av målen står i konflikt med (motverkar) varandra agerar företaget här på så sätt att satisfieringsnivåerna för samtliga ultimärmål söks uppnås. Situationer kan föreligga då mål på lägre nivåer är precedenter till flera ultimärmål samtidigt.

För det fall målkonflikter mellan satisfieringsmål på ultimärnivå skulle vara aktuella blir analys av konfliktternas natur nödvändig. Om ultimärmålen är operationella, bör konfliktsituationen i vissa fall vara hanterbar. Låt oss exempelvis tänka oss att ett företag samtidigt strävar mot två ultimärmål A och B, båda av satisfierande typ, samt att ett bidrag till uppfyllandet av A också samtidigt innebär ett negativt bidrag till uppfyllandet av B. Ökar vi uppfyllelsen av A, så motverkar vi alltså därmed uppfyllelsen av B. För den händelse den motverkande effekten är känd till sin omfattning är det därvid möjligt att, om ej att uppnå både A och B samtidigt, så åtminstone att beräkna, och sträva mot, den uppfyllelsenivå som innebär minsta totala avvikelse från A och B. Det bör i denna situation alltså finnas ett "jämviktsläge", där de båda målen samtidigt uppnås "minst dåligt". Tanken kan extrapoleras till fler än två motstridiga mål.

Ett ultimärmål är av optimerande typ, och de övriga satisfierande.

I denna situation strävas uppenbarligen mot att uppnå satisfieringsnivåerna för de icke-optimerande målen, samtidigt som uppfyllelsen för det optimerande ultimärmålet maximeras. Skulle konflikter föreligga mellan de satisfierande målen, kan handlingssättet från närmast ovanstående appliceras. Är däremot det optimerande målet berört av konflikt med de övriga målen, kan formell metodik ej appliceras för måluppfyllelsen. Härvid tvingas företaget till beslutsfattande av intuitiv natur, en situation som ej är sällsynt i praktiken.

Flera ultimärmål är av optimerande typ.

För det fall inga positiva eller negativa samband råder mellan ultimärmålen erhålles total måluppfyllelse genom maximering av uppfyllelsen för vart ultimärmål för sig, s k Pareto-optimering. Då emellertid två eller flera av målen delvis (eller helt) motverkar varandra synes intuitiva beslut vara tvingande aktuella. Formell bearbetningsmöjlighet av denna multimålsituation i syfte att lösa upp konflikterna kan knappast anges.

Langefors belyser i (16), sid 1 samt 3, denna situation:

> "It must be emphasized that the multi-goal situation means that decisions cannot be made in a totally rational way".
> "The fact that there is no longer a single objective, goal, implies that there is not - in general - a single objectively best solution".

Hur härvid styrningen tvingar till intuitiva beslut kommer något att belysas nedan.

4.4 Målkonflikter i multimålsituationer

Vi har ovan noterat möjligheter till konflikter mellan mål på ultimärnivå. Givetvis kan målkonflikter tänkas aktuella även på lägre nivåer.

Trots att generella utsagor knappast kan göras beträffande detaljutseende hos olika företags målstrukturer kan vi belysa konfliktsituationer genom studier av tillgänglig litteratur. Det vill förefalla som om en "modern" ståndpunkt avseende målanalys bygger på följande antaganden:

1 Företag styrs oftas mot fler än ett mål. Underlåtenhet att redovisa fler än ett mål innebär vanligen att analysen icke beaktat tillräckligt antal påverkande faktorer.

2 Konflikter på ultimärnivå (eller på lägre nivåer) är vanligt förekommande, trots att så ej alltid uppmärksammas av de aktuella företagen.

Johnsen ägnar huvuddelen av analysen i (10) åt problemen kring hantering av multi-mål-situationer. I motsats till "tidigare" förekommande utsagor om t ex vinstmaximering som enda företagsmål konstateras i (10), sid 29:

> "Our analysis of the literature shows that an increasing number of researchers tend to hold the view that profit maximization is not an adequate expression of the goals of a firm".

Utgående från det uppenbara att unimålsituationer kan ses som ett specialfall av multimålsituationen (vilket dock knappast framhävs tillräckligt i (10)) anger Johnsen vidare i (10), sid 28 följande allmänna konstaterande:

> "It is meaningful to describe goal conditions in a firm by means of a multigoal model".

De önskemål om rörlighet i måluppfyllelsen som ofta blir aktuella i multimålsituationer där individer verkar, belyses av Marin-Norman i (14), sid 365:

> "We expect and tolerate regulation and assume that it will be in our general interest. At the same time we require freedom - to pursue with minimum restraint those individual ends we hold most precious.
> We pay a price for our freedom. The cost lies in not optimizing our total effort in pursuit of a single goal. But that price can be offset against the probability that the single goal is not the one most worth pursuing. Potentially, we have at our disposal, in modern management science, techniques that can be applied to a whole nation in the same way they are applied to a single firm. In deciding how we shall put them to use we must decide where to draw the line between the freedom of the citizen and the efficiency of the employee".

Ett under 70-talet sannolikt permanent aktuellt exempel på konflikter i multimålmiljö rör de motverkande krafterna mellan bevarande av en orörd naturmiljö och upprätthållande av en hög sysselsättning, speciellt i landsdelar med hittills lågt utnyttjande av naturresurserna, samt med gles befolkning. Dylika målkonflikter kan hanteras genom att uppge delar av uppnåelsen av det ena målet till förmån för det andra.

Konfliktsituationer inom företag behandlas vidare av Hughes i (17), sid 13. Det framhålles där:

> "Can a company and its employees both have what they want, or is dissatisfaction inevitable for either side? This question - so often asked today - sums up one of the chief problems facing our free enterprise economic system in a era of growing controls on behavior: how to make the organization and its individuel members more effective. The answer would seem to lie in harmony of goals and in a concept of goal setting which will motivate individuals to work together both for the common cause and for personal reasons".

Vi har noterat att dylik "harmony of goals" (syftande på önskemål) icke alltid är möjlig uppnå. Vi har också konstaterat att det i en multimålsituation med konflikt mellan två eller flera av önskemålen, blir nödvändigt acceptera en nivå vid vilken bidrag till uppfyllelsen av målen är begränsad.

4.5 Intuition i beslutsfattande

I tidigare resonemang har befunnits att det gives situationer då beslutsfattande icke kan formaliseras (exempelvis införas, programmeras, in i en dator). En sådan situation föreligger exempelvis då ett beslut måste fattas som berör flera motstridiga, icke-operationella optimerande mål. Situationen antas kräva ett beslut. Beslutsfattaren har därvid endast intuition, förnuft eller omdöme att förlita sig på vad gäller val mellan olika alternativ. (Vi utgår från att tillräckligt med sakinformation om beslutsalternativen och dessas konsekvenser finns till hands.)

Det faktum att situationer kan uppkomma, och uppkomma frekvent, då intuitiva beslut är tvingande nödvändiga, medför att vi inte kan döma ut intuitionen som beslutsmedverkande faktor.

Det kan antas att intuitiva beslut baserar sig på kalkyler i hjärnan på beslutsfattaren, dock att dessa kalkyler sannolikt icke låter sig formaliseras explicit på ett för omvärlden begripligt sätt (ännu?).

Vi kommer senare att diskutera hur, och i vilka situationer mål kan vägas mot varandra. Langefors ger i (9), sid 95, en belysning till intuitionens användbarhet i samband med dylik "viktning" av mål (utsagon nedan är sannolikt giltig i större sammanhang än det nämnda):

> "När man, på grund av den ofrånkomliga grovheten i viktbestämningarna, avvisar tanken på deras användning, och i stället använder eller förordar direkta, intuitiva beslut glömmer man att dessa innebär att en totalviktning de facto göres - man accepterar då en ännu grövre mätning - kanske därför att man då ej inser att den utgör en mätning".

I många fall är det sannolikt att intuitionen baserar sig på

prioritering, en situation då vikten för ett mål sätts mycket hög, eller vikten för konkurrerande mål avsiktligt sätts mycket låg. Vi ser här ett kvantitativt samband mellan viktning och prioritering.

Johnsen redovisar i (10), sid 362 ff, flera teorier angående människans förmåga att hålla beslutsinformation aktuell i minnet, vilket kan anses utgöra en grund för intuitiva beslut. Vi nämner därur här endast

> "the conclusion that behavior is normally related to more than one motive, and therefore to more than one goal".

Det noteras därefter, under samma rubrik, som en slutledning att:

> "man is able cognitively to handle a multitude of goals".

Det vill synas uppenbart att så är fallet. Verkligt intressant blir problemet dock först när frågan ställs ang hur väl människan kan fatta beslut i multimålsituationer. Vi lämnar dock denna frågeställning, då den vill synas mera perifert beröra föreliggande ämne, och snarast bör höra hemma inom psykologisk/sociologisk forskning.

Avrundningsvis kan vi, med Hughes i (17), sid 59, konstatera att det förefaller klokt att i möjligaste mån sammanjämka formella och intuitiva beslutsmetoder, och:

> "strive for balance between the long-range goals of the individual and those of the organization"

vid applicerandet av metoderna. Langefors formulerar i (9), sid 55, detta som:

> "Om vi kan utveckla en ny syn på användningen av formell metod, som ersätter strävan att göra sig oberoende av intuition - efter en insats därav i uppläggningsskedet - med en strävan att finna ett bra sätt att koppla in intuitiv bedömning i vissa situationer där detta är viktigt, så torde den formella metoden bli det viktiga styrningshjälpmedel man har föreställt sig".

4.6 Kvantifiering av mål

En betydande svårighet vid all målanalys utgöres av det besvärliga i att finna alla mål som är faktiskt aktuella för företagets verksamhet. Det är sannolikt att även den mest noggranna och välsyftande målanalytiker i praktiken kommer att förbise vissa delmål (kanske rent av önskemål) som har betydelse för företaget. Detta påstående (förmodande) låter sig svårt bevisas, bl a brist på empiriskt erfarenhetsmaterial, men bör trots detta ej underlåtas att framhållas.

I de fall det ovan givna förmodandet är sant, följer att formaliserad totalstyrning för företaget knappast kan bli aktuell. Det är då motiverat att sikta mot formaliserad styrning endast av de företagsdelar, för vilka fullständigheten i målanalysen kan garanteras. (Åtminstone med en ej för liten sannolikhet).

Formaliserad styrning av avgränsade företagsdelar kan uppenbarligen vara möjlig, och f ö även eftersträvansvärd. För att dylik skall kunna genomföras bör de för den formella styrningen aktuella målen kvantifieras.

Vi har tidigare diskuterat att i företagets målstruktur precedensförhållanden kan råda mellan de i strukturen ingående målen. De delar av strukturen där samtliga mål är att betrakta som operationella, kan utväljas och göras till kandidater för formaliserad styrning. I ett dylikt delsystem må ingående mål vara satisfierande eller optimerande, dock kan vanligen satisfierande mål hanteras "med större säkerhet". Detta kan påstås enär satisfieringen innebär existensen av ramar inom vilka måluppfyllelsen skall hamna, ramar som i sig skall vara mätbara med tillräcklig noggrannhet. Ofta utgöres en dylik ram enbart av intervallet över en viss gräns (tröskel), varvid uppfyllelsen av målet innebär endast att den nämnda tröskeln skall passeras. Även existensen av en övre gräns för måluppfyllelsen kan emellertid tänkas.

I målstrukturen i fig 4.6:1 har vi inringat ett delsystem av operationella mål, vilket vi avser kvantitativt diskutera i samband med formaliserad styrning.

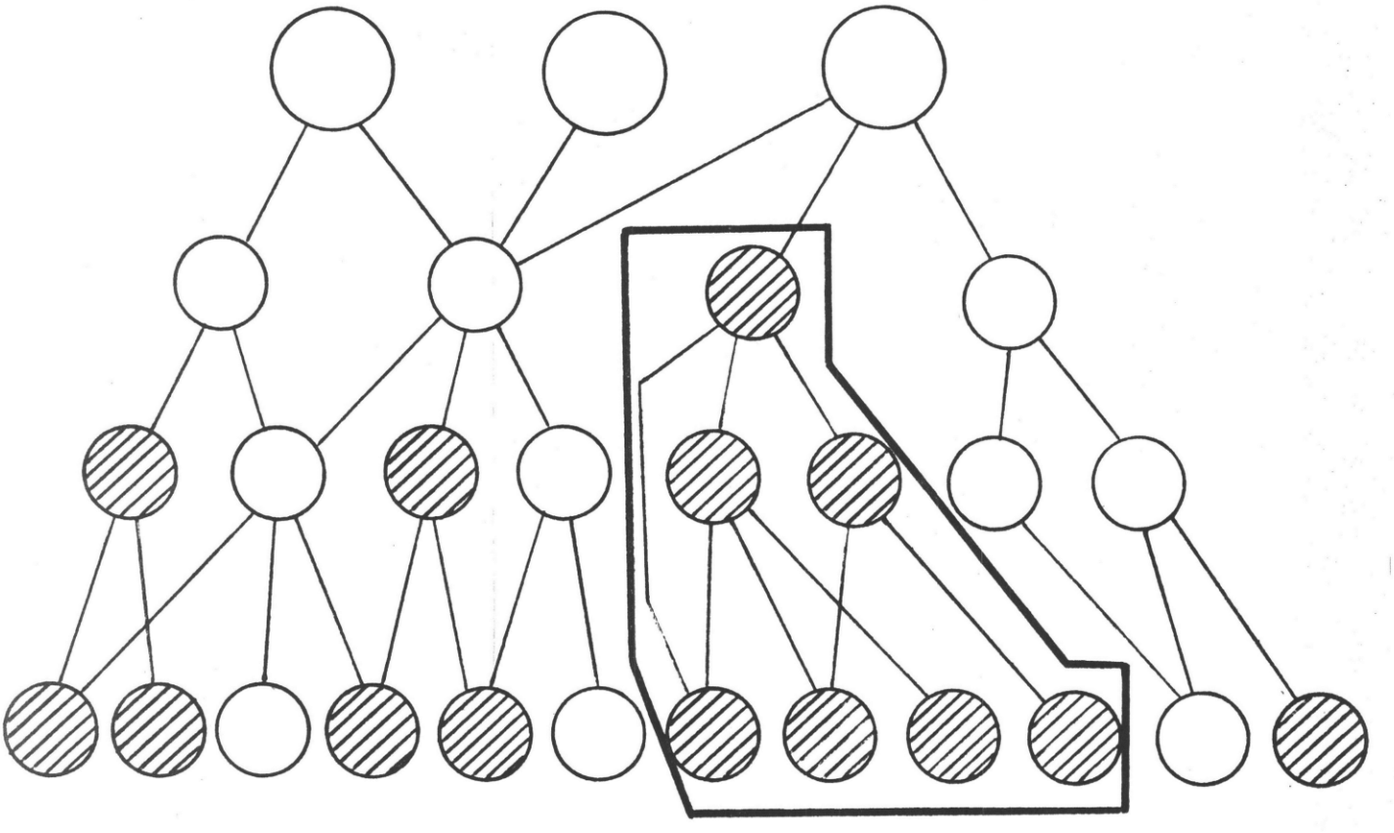


Fig 4.6:1 Operationell del av målstruktur
(streckade mål är operationella och satisf.)

Frågan är nu under vilka förutsättningar vi kan tänka oss att sätta vikter på målen i detta delsystem, för att i och med en viss måluppfyllelse på lägsta (operativa)nivå kunna konstatera eventuell måluppfyllelse högre upp i strukturen. Det förefaller klart att dylik viktning ej generellt kan tillämpas. Vi kan nämligen tänka oss situationer då vägningen är beroende av uppfyllelsen av ett eller flera i strukturen ingående mål. En sådan situation föreligger t ex när det anses att en ökning av uppfyllelsen av ett visst mål direkt innebär en minskning av det ifrågavarande målets vikt. I de fall inga dylika restriktioner är aktuella, kan emellertid ansättande av vikter på mål vara användbart. Vi bör emellertid observera att vikter, såväl som mål själva, ofta är att betrakta som dynamiska, varvid de kan anses giltiga endast för viss tidsrymd. Även annan påverkan kan naturligtvis vara aktuell. En "generell" viktning, gällande under alla förutsättningar och för lång tid, är sannolikt orealistisk. Dessutom bör påminnas om att all viktning, liksom målansättning i allmänhet, är subjektiv. Någon allmän "vägningsmoral" kan sålunda knappast åberopas.

Två typer av viktning kan tänkas: absolut och relativ. Den absoluta viktningen innebär angivande av ett (absolut) viktmått för varje mål, oberoende av andra på samma strukturnivå ingående mål. Vid relativ viktning ges emellertid vikter under hänsynstagande till "bredvidliggande" mål. Relativ viktning är sålunda betydelsefull för avvägning mellan mål, medan den absoluta viktningen kan vara värdefull för avvägningar avseende resursbehov för måluppfyllelse. Vi kan se det så att relativ viktning "verkar horisontellt", medan den absoluta viktningen "verkar vertikalt" i målstrukturen.

För analys av målstrukturer med absoluta värden för måluppfyllelse, i enligt med vårt sätt att betrakta målstrukturer som system, kan systemalgebraisk metodik komma i fråga. Langefors har i (6) som en del av systemalgebran

angivit en metod för "behovsräkning" som kan användas vid målanalys. Vi genomgår här inte bakgrunden till denna metod, utan hänvisar till (6), och belyser enbart med ett exempel.

Delsystemet i fig 4.6:1 avbildas nedan:

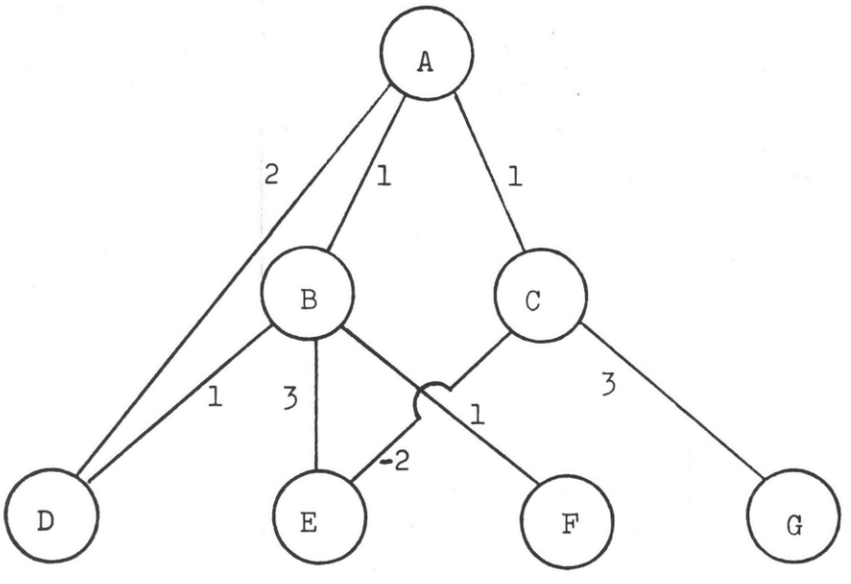


Fig 4.6:2 Exempel på målbehovsanalys

I figur 4.6:2 har vi antagit att det vid förberedande målanalys framkommit att för att uppfylla(det operationella) satisfieringsmålet A det krävs "tvåfald" (dubbel) uppfyllelse av satisfieringsmålet D, samt dessutom "enkel" uppfyllelse av satisfieringsmålen B och C. På motsvarande sätt anges i figuren fordringarna för målen B och C, uttryckta i D,E,F och G. Vi ser t ex att uppfyllelse av målet E trefalt bidrar till uppfyllelse av målet B, men samtidigt tvåfalt motverkar uppfyllelse av målet C. Vi har här exempel på en faktisk målkonflikt.

Antag att för uppfyllelse av ett mål på en högre nivå än A det krävs trefald uppfyllelse av A. Vilken uppfyllelse av B, C, D, E, F och G innebär detta?

Systemets numeriska precedensmatris uppställs, samt selektionsvektor s^o:

$$P^{oo} = \begin{array}{c|ccccccc} & A & B & C & D & E & F & G \\ A & & & & & & & \\ B & 1 & & & & & & \\ C & 1 & & & & & & \\ D & 2 & 1 & & & & & \\ E & & 3 & -2 & & & & \\ F & & 1 & & & & & \\ G & & & 3 & & & & \end{array} \qquad s^o = \begin{array}{c|c} & \\ A & 3 \\ B & \\ C & \\ D & \\ E & \\ F & \\ G & \end{array}$$

Att systemet saknar cykler framgår här av att alla element i precedensmatrisen över huvuddiagonalen är noll.

Beräkningen fortlöper enligt nedan:

$$S^1 = P^{oo} \cdot s^o = 3 \cdot \begin{array}{c|c} & A \\ A & \\ B & 1 \\ C & 1 \\ D & 2 \\ E & \\ F & \\ G & \end{array} = \begin{array}{c|c} & A \\ A & \\ B & 3 \\ C & 3 \\ D & 6 \\ E & \\ F & \\ G & \end{array}$$

$$S^2 = P^{oo} \cdot S^1 = 3 \cdot \begin{array}{c|c} & B \\ A & \\ B & \\ C & \\ D & 1 \\ E & 3 \\ F & 1 \\ G & \end{array} + 3 \cdot \begin{array}{c|c} & C \\ A & \\ B & \\ C & \\ D & \\ E & -2 \\ F & \\ G & 3 \end{array} + 6 \cdot \begin{array}{c|c} & D \\ A & \\ B & \\ C & \\ D & 3 \\ E & 3 \\ F & 3 \\ G & 9 \end{array} = \begin{array}{c|c} & A \\ A & \\ B & \\ C & \\ D & 3 \\ E & 3 \\ F & 3 \\ G & 9 \end{array}$$

$S^3 = P^{oo} \cdot S^2 = \emptyset$ (inga element $\neq 0$ i kol D, E, F och G)

Totalbehovet $s^{tot} = \sum_{i \geq 0} s^i =$

3 A
3 B
3 C
9 D
3 E
3 F
9 G

Alternativt hade vi kunnat beräkna s^{tot} som $P^{oo^{tot}} \cdot s^o$.

Vi finner alltså att för den trefalda uppfyllelsen av A krävs trefalt B, trefalt C, niofalt D, trefalt E, trefalt F och niofalt G.

Ovan har antagits att ingående satisfieringsmål var av tvåvärd (binär) art, d v s de kunde endast vara "uppfyllda" eller "icke uppfyllda". Vi kan expandera detta betraktelsesätt, och betrakta situationen att över en viss satisfieringströskel flera värden för måluppfyllelsen kan vara aktuella. Vi kan se detta som ett mellanting mellan begreppen satisfiering och optimering, dock med tonvikt på det förra.

Viktsättning av mål besitter den fördelen att, så länge vikterna är giltiga, matematisk kalkyl kan användas i samband med styrning mot målen. Härvid införs inga ytterligare osäkerhetsmoment under kalkyleringen.

Å andra sidan har vi konstaterat att vägning endast bör användas i vissa situationer. I andra fall kan prioritering, d v s successivt (ett-efter-ett) betraktande av olika mål som primärmål, vara den framkomliga vägen. Vi har dock nämnt att detta kan ses som ett specialfall av vägning (med "oändlig" resp noll-vikt).

4.7 Konstruktion av målstrukturen

Vid konstruktion av företagets målstruktur kan principiellt olika ansatser göras: start uppifrån (med ultimärönskemålen) eller start nedifrån (med de operativa målen). Vilken ansats är lämpligast? Som med allt systemarbete, och konstruktion av målstrukturer är alltså att betrakta som ett sådant, är iterationer nödvändiga. Vi kan alltså inte räkna med att under alla förutsättningar nå denslutliga strukturen direkt genom en viss, given ansats. Iterationer kommer att vara nödvändiga. Det är därför snarast fråga om att välja en ansats som minimerar antalet erforderliga iterationer. Detta optimeringsproblem är svårlöst, då skälen för itereringen i praktiken är starkt varierande från fall till fall. Bl a kan på förhand ej sägas säkert hur långt ner operativa mål bör ansättas. Det kan därför tänkas att med Granger (i Miller: Objectives and Standards, sid 34, ref i Langefors (9), sid 140) säga:

> "You start by setting objectives wherever you can - in the middle, at the top, at the bottom .... The important thing is to get started".

Då vi eftersträvar en viss formalisering av målanalysarbetet är emellertid en sådan löslig arbetsmetodik mindre rekommendabel. Erfarenheten talar för det fördelaktiga i att starta arbetet med målangivelserna uppifrån. Det är lämpligt att för att minimera riskerna att inte få med alla väsentliga mål, så många som möjligt av företagets intressenter bör tillfrågas angående deras uppfattning om målmängdens utseende.

Vid undersökning av målsamband i strukturen kan precedensanalys (se Langefors (6)) vara ett värdefullt hjälpmedel för att öka hanterbarheten hos strukturen.

Langefors anger i (9), sid 75:

> "Det är uppenbart att långtgående analogier med informationssystemen och informationsprecedensanalysen råder vad beträffar önskemål och nyttokomponenter och deras precedensrelationer".

Som vi tidigare diskuterat är det klokt att i så stor utsträckning som möjligt söka ansätta operationella, satisfierande mål. Målvärden för uppfyllande av dessa mål kan, som vi sett i 4.6 analyseras med hjälp av behovsanalys. Härvid kan kvantitativa samband (inklusive konflikter) mellan startmål, mellanliggande mål och slutmål belysas.

5 Databehandlings-systemegenskapernas natur

5.1 Egenskapernas kännetecken

Vi övergår nu till att betrakta naturen, eller arten, av de egenskaper som känetecknar ett datorsystem. Vi måste få stabilt grepp om dessa egenskaper för att kunna bedöma deras möjliga anpassningsförmåga till företagets operativa mål. Det finns skäl att genomföra denna betraktelse på två plan. Inledningsvis vill vi fästa tonvikt vid ordet system i ett datorsystem, och allmänt diskutera anpassbarheten för detta system. Senare, efter diskussion av utvärderingsmetodik, kommer vi att mera i detalj studera egenskaper som kännetecknar just datorsystem, och hur dessa kan behandlas.

Bakgrunden till denna uppdelning av diskussionen är att datorsystemen på senare tid (70-talet) kan sägas ha nått en sådan kraftfullhet i användningsförmåga och anpassbarhet att de med fog kan sägas kunna utgöra en fundamental bearbetande del av ett företag. Det är knappast längre primärt fråga om att hålla ett antal knåpande fackdataspecialister sysselsatta inom en huvudsakligen PR-motiverad dataavdelning inom företaget. Betydelsen av och möjligheterna för databehandlingen har numera nått längre än så, många företag är i dag helt beroende av denna databehandling, på samma sätt som de är beroende av andra viktiga produktionsavdelningar. Databehandlingens betydelse synes därför motivera att dess karaktäristika belyses dels 'systematiskt' och dels successivt mera i detalj.

Företagets databehandlingssystem utgör en fundamental del av företagets informationssystem. Skälet för att söka efter ett "förbättrat" databehandlingssystem (där vårt utvärderingsproblem kommer in i bilden) är att vi önskar modifiera eller omskapa informationssystemet till att i högre grad uppfylla företagets mål. Detta kan vi åskådliggöra i följande figur:

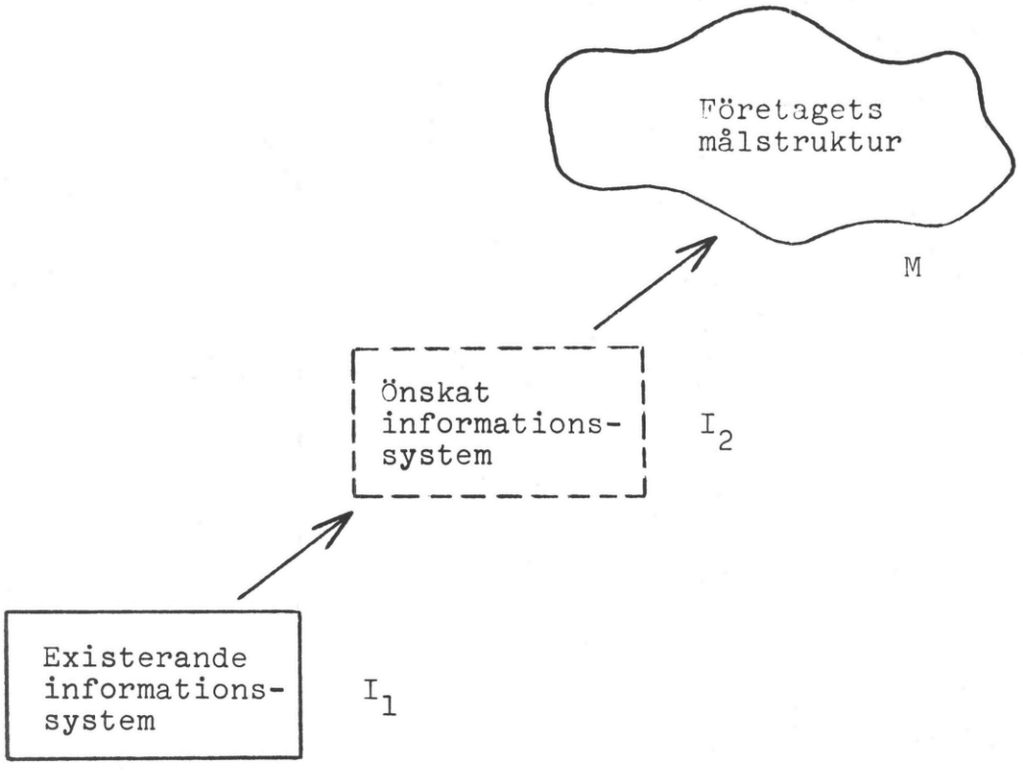


Fig 5.1:1

I denna figur observerar vi flera problemställningar. Låt oss anta att företagets målstruktur M är definierad och preciserad tillräckligt i detalj. Låt oss dessutom anta att det existerande informationssystemet I_1 har analyserats, och dess status är väl definierad. Vad avser vi med "väl definierad" här? Vi kan tänka oss någon sorts skala applicerad på systemet, eller rättare sagt ställa systemet i relation till en sådan skala. Vi kan även se denna skala som en koordinataxel, relaterad till vilken vi givit I_1 ett mätvärde eller en koordinat. Vad är det vi därmed har uppmätt? På detta allmänna resonemangsplan kan vi säga att vi bl a då har betraktat ambitionsnivån för systemet.

Det är svårt att precisera vad vi avser med begreppet "ambitionsnivå" utan att gå in närmare på hur aktuella system är utformade. Grovt kan dock sägas att vi i begreppet innefattar bl a systemets

- generalitet
- flexibilitet
- styrbarhet

Vi kan nu utöka vårt resonemang med två dimensioner, och tänka oss tre koordinataxlar, var och en med en skala för en av de tre nämnda systemegenskaperna. Har vi fastslagit mätvärden för I_1 m a p dessa tre egenskaper har vi då angivit ambitionsnivån som en punkt i den av de tre axlarna uppspända tredimensionella rymden. Själva ordet ambitionsnivå indikterar emellertid att vi avser en nivå, vanligen åskådliggjord som ett plan i en given rymd. Vi skulle givetvis kunna konstruera en rymd där ambitionsnivån åskådliggjordes som ett plan, men nöjer oss här med att betrakta den som en punkt enligt ovan.

Det <u>önskade</u> informationssystemet I_2 <u>förmodas</u> befinna sig någonstans närmare målstrukturen M än I_1. Vi kan tolka detta som att ambitionsnivån för I_2 antas "större" än ambitionsnivån för I_1. Med beteckningen A för ambitionsnivå, applicerbar som funktion på motsvarande informationssystem, kan vi formulera detta som:

$$A(I_2 > A(I_1)$$

Det är nu av vikt att notera att $A(I_2)$ inte är på förhand känd av företagsledningen. En av de betydelsefulla arbetsuppgifterna för företaget är i stället att <u>välja</u> lämpligt värde på $A(I_2)$. Hur skall detta gå till? Emedan olika I_2 ännu ej existerar kan t ex ej $A(I_2)$ empiriskt mätas för tänkbara situationer, varefter lämpligt värde på $A(I_2)$ kan fastslås. Vi skulle behöva känna resursbehovet för olika I_2 för att kunna bestämma oss för vilket (vilken nivå) vi skall satsa på.

Tyvärr är ofta fallet att $A(I_2)$ är beroende av den erforderliga arbetsmängden för förändring från I_1 till I_2. Till exempel, om i en (förenklad) företagssituation endast målet "kostnadsminimering" skulle vara aktuellt, är det rimligt att till $A(I_2)$ knyta kostnader för förändringen från I_1 till I_2. Langefors formulerar i (18) ovan nämnd problemställning:

> "Before design can be done, the ambition specification must be decided upon, and this requires knowledge of the cost of the design which, of course, can only be determined after the design. Thus, a strong requirement for iterative treatment is indicated".

Vi finner alltså situationen vara den att för att kunna välja en lämplig systemutformning (eller åtminstone ambitionsnivån för densamma) behöver vi känna till detaljer om det färdiga systemet. Detta problem kan inte omedelbart lösas explicit, utan vi tvingas alltså till ett iterativt förfarande. Det blir fråga om att mer eller mindre intuitivt ansätta ett startvärde för $A(I_2)$, greppa tillräckligt väsentliga delar av I_2 (för att få grepp om systemresursbehovet), modifiera $A(I_2)$, greppa detta till väsentliga delar o s v.

Den iterativa arbetsgången som blir nödvändig är ofta i praktiken besvärlig att realisera. Det från starten något så när intuitivt ansatta utgångsvärdet för $A(I_2)$ blir ofta normerande, bl a på grund av att storleken på den för systemrealiseringen erforderliga arbetsmängden, i många fall ekvivalent med systemrealiseringskostnaden, ofta visar sig bli större än väntat, därmed begränsande den påföljande rörligheten. Det kan uttryckas förhoppningen att större vikt i framtiden skall fästas vid systemens flexibilitet än som hittills ofta varit fallet. Därmed skulle iterationer i det här aktuella avseendet ha större utsikter att verkligen kunna komma till stånd.

Vi bör inte underlåta att fästa uppmärksamhet på att för att dessa iterationer meningsfullt skall kunna verkställas, måste tillräckligt exakta mätmetoder för systemens utfall kunna appliceras. Vi måste kunna notera vart vi skall iterera. Det har visat sig att i många hittills implementerade system man nöjt sig med att efter implementeringen glädja sig åt att systemen över huvud taget fungerar. Naturligt hade varit att i större utsträckning än som visat sig bli fallet istället fråga sig hur väl systemen fungerar, d v s att lägga ner arbete på att mäta hur väl de kommit att uppfylla inledningsvis ansatta specifikationer (hur nära målen de kommit).

5.2 Konkretisering av databehandlingssystemets ambitionsnivå

Vi har ovan funnit att diskussion om databehandlingssystemets allmänna egenskaper nära sammanhänger med diskussion om systemets ambitionsnivå. Vi utvecklar nedan dessa tankar ytterligare.

På en grov nivå kan databehandlingssystemets egenskaper för företaget sammanföras till:

1 Grad av service åt företaget

2 Resursbehov

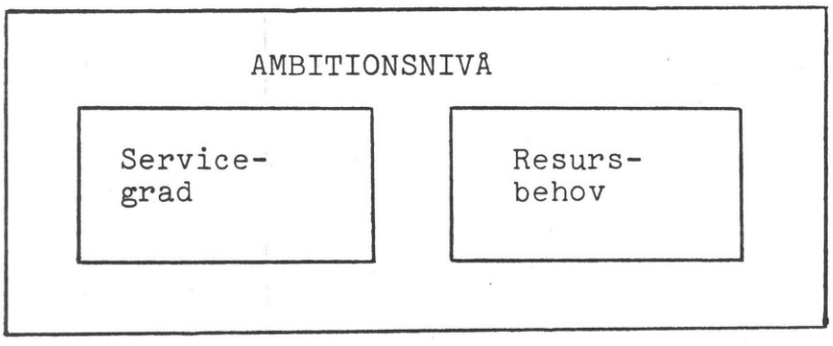


Fig 5.2:1

Dessa två delar är normalt kopplade till varandra på så sätt att en ökning av servicegraden innebär en ökning av resursbehovet (d v s proportionalitet föreligger).

På vilket sätt kan vi närmare spjälka upp servicegranden ovan? Vi kan anföra följande uppdelning:

1 Generalitet. Häri innefattar vi dels generalitet hos systemets 'kärna', dels systemets omfång, vilken detaljrikedom det innebär. Vi kan notera att med det senare avses den detaljrikedom som är av konkret värde, d v s

som kan utnyttjas för att förbättra måluppfyllelsen. (Varje konkret system kan anses besitta stor detaljrikedom, på t ex kemiskt atonär nivå. Atomer kan dock normalt inte direkt utnyttjas för att förbättra måluppfyllelse. Det är alltså inte sådan detaljrikedom som här avses.)

2 Flexibilitet. Här betraktar vi de möjligheter som systemet besitter för att kunna hanteras (eller klara sig) inför förändringar i omgivande miljö. Dylika förändringar (som vi avseende mål tidigare har diskuterat) uppkommer ständigt, och det kan anses som en faktor av ökande betydelse att företagets system kan arbeta i en föränderlig omgivning.

3 Styrbarhet. Nära sammanhörande med flexibilitetsbegreppet befinner sig systemets styrbarhet. Styrbarheten avser egenskaper med vilkas hjälp vi direkt eller indirekt kan manövrera systemet (medan flexibilitet kan anses innefatta även självanpassbarhet). Styrbarheten hanteras med hjälp av ett antal konkret åtkomliga styrvariabler (se t ex Langefors (9), sid 29) som kan fås att påverka systemets uppförande.

4 Struktur. Vi inkluderar här egenskaper avseende systemets uppbyggnad, d v s vi avser med ett välstrukturerar system ett system med väl definierade delsystem och avgränsade kopplingar mellan ingående delsystem.

Det kan tänkas flera delegenskaper inom begreppet servicegrad. Vi nöjer oss emellertid här med de ovan givna.

I samband med val av lämplig ambitionsnivå för databehandlingssystemet måste resursbehovet för att förverkliga och driva systemet beaktas. Vi innefattar här i 'resurser' alla tillgångar för företaget som förbrukas i samband med konstruktion och drift av det nya systemet, således icke endast penningkostnader. Uppenbart är att den resursförbruk-

ning som krävs av ett nytt system måste relateras till företagets mål. Inget företag har obegränsade tillgängliga resurser. Det är därför naturligt att ett av målen för de flesta företag är att antingen hålla resurskraven under en given nivå (satisfiering) eller att minimera dessa krav (optimering). Val av ambitionsnivå kan därför även ses som ett fastställande av vissa operativa mål, sådana som vi beslutar oss för att eftersträva uppnåelse av, och som berörs av databehandlingssystemet.

Vi har här ännu ej berört begreppet produktivitet. Detta kan anges som förhållandet mellan producerat resultat och använda resurser. Snarast kan vi se det 'producerade resultatet' som jämställt med gjort val av ambitionsnivå, varför denna nivå kan betraktas som ingående i täljaren i produktivitetskvoten. Att val av ambitionsnivå är relaterat till målsättningen framgår därmed även ur det förhållandet att en viss produktivitetsnivå sannolikt ingår någonstans i målstrukturen för de flesta företag. Detta samband formulerar Johnsen i (10), sid 344, som:

> "If no aspiration level exists in relation to a certain behavior, this behavior is considered to be nongoal-directed".

Val av lämplig ambitionsnivå för databehandlingssystem kan, som en sammanfattning av ovanstående, ses som en viktig del av systemering, en del som tyvärr ofta inte explicit kan särskiljas från övriga systemeringsavsnitt. Detta på grund av det iterativa arbetsförlopp som all systemering kräver. Vi citerar här Langefors (8), sid 2-3:

> "... all systemering får börja med en översiktsanalys av hela systemet och därefter mera specifikt studera en serie av ändringar eller tillägg av väldefinierade delsystem. Vilka ändringar eller tillägg som skall genomföras, eller hur ambitiösa de skall vara, alltså vilken ambitionsnivå som skall väljas, blir en fundamental målsättningsfråga för systemeringen. Den måste avgöras på

basis av jämförelse mellan det nyttovärde varje ändring kan åstadkomma (genom bättre företagsresultat totalt) och den kostnad som ändringen medför".

Vi noterar att valet av ambitionsnivå knappast i sin helhet kan företas före målanalysen. Vid systemets planering bör detta förhållande beaktas.

5.3 Nedbrytning av databehandlingssystemets egenskaper

I vårt ovan förda resonemang har vi behandlat databehandlingssystemets egenskaper på en grov nivå, men antytt de uppenbara möjligheterna att bryta ned, eller uppspjälka, dessa egenskaper ytterligare. För att kunna diskutera anpassning av databehandlingssystemet till företagets mål, vilka vi förmodar nedbrutna till operativ nivå, krävs uppenbarligen egenskapsnedbrytning av databehandlingssystemet. Vi kan omöjligtvis direkt ta ställning rörande t ex datorsystemutformning och/eller fabrikat utan att denna kompatibilitet mellan system och mål föreligger.

Som vi tidigare noterat innebär utvärderingen att ta ställning i frågan rörande "bästa" val av databehandlingssystem. Problemområdet är emellertid, allmänt sett, inte omedelbart beroende av hur aktuella databehandlingssystems egenskaper ser ut. Det är tillfyllest med att konstatera detta systems nedbrytbarhet till ett antal mer eller mindre elementära egenskaper, för att kunna diskutera anpassningen. Vi väljer här den arbetsformen att nu närmast betrakta själva utvärderingens innehåll, och kommer därefter att återkomma till det explicita utseendet av databehandlingssystemets tekniska egenskaper. Därvid kommer bl a systemens prestanda att belysas.

7.3 Formell sammanställning av egenskaper

Den i det enskilda fallet lämpliga formuleringen av frågor (bl a i offertsammanhang) för att belysa aktuella systemegenskaper är självfallet i hög grad beroende på avsedd användningsmiljö för systemet. På en grov nivå kan emellertid alla system - liksom vi diskuterat i samband med begreppet ambitionsnivå - sägas ha egenskaper som kan sammanföras under vissa allmänna rubriker. Vi anger nedan några sådana rubriker, under vilka aktuella systemegenskaper från lägre nivåer kan sammanställas. (Listan nedan är givetvis ej fullständigt komplett).

1. Flexibilitet
   Hit hänförs egenskaper hos systemet som har betydelse för dess förmåga att vara anpassbart till en (ofta framtida) varierande miljö.

2. Effektivitet
   Här hör egenskaper hemma som anger systemets prestanda avseende olika enstaka mätvariabler i aktuell omgivning.

3. Generalitet
   Hit hänförs egenskaper som sammantaget anger systemets möjligheter att fungera i vida klasser av problemsituationer.

4. Reliabilitet
   Under denna rubrik noteras egenskaper som anger systemets pålitlighet och/eller säkerhet avseende aktuella tillämpningar.

5. Hanterbarhet
   Här noteras egenskaper som visar systemets "människovänlighet" och förmåga att kunna styras efter givna yttre (ofta mänskliga) konventioner.

6. Kompatibilitet
   Här noteras sådana egenskaper som belyser systemets förmåga att kunna exakt anpassas till givna systembetingelser.

7. Resursförbrukning
   Hit hänförs de egenskaper som anger systemets förbrukning av de resurser som dess drift kräver.

6 Metodik för utvärdering

6.1 Beskrivningsprinciper

Våra hittills förda resonemang har gjort det antagligt att det är möjligt att bryta ner såväl företagets målstruktur som önskat datorsystems egenskaper till en nivå där (viss) strukturell kompatibilitet råder mellan de två. Vi har dock ännu inte närmare definierat denna nivå. Detta kommer vi senare att diskutera i samband med diskussion av utsändande av offertförfrågningar.

Utvärderingen innebär att vi med lämplig metod fastslår vilket av ett antal tänkbara databehandlingssystem, som "bäst" uppfyller målen, d v s vilket system som ger största bidraget till ökningen av måluppfyllelsen. För att kunna fatta beslut härom krävs inte bara strukturell kompatibilitet mellan mål och egenskaper, utan även beskrivningsmässig överensstämmelse. Det är önskvärt att systemets samtliga egenskaper kan beskrivas i samma skalenhet, en enhet som kan relateras till de till operativ nivå nedbrutna målen.

Inkonsistenser som uppkommer om inte systemegenskaperna beskrivs i samma enhet, exemplifieras i Johnsen(10), sid 431, där det visas hur en vägd summerings resultat är direkt beroende av skillnader i olika skalenheter.

Erfarenheter talar för att det är synnerligen svårt, om inte omöjligt, att finna en dylik fysikalisk enhet, giltig för alla typer av datorsystem och för alla typer av målstrukturer. Langefors behandlar detta problem i (9). Systemegenskaper för datorsystem kan i bästa fall uttryckas i enheter av typen

- antal bearbetade jobb per tidsenhet
- användarvänlighet
- resursförbrukning för systemets drift, uttryckt i mankraft (manpower)
- resursförbrukning för systemets drift, uttryckt i lämplig penningenhet

m m.

Dessa enheter är sinsemellan synnerligen artskilda, och i de flesta fall även skilda från enheter, i vilka den operativa måluppfyllelsen beskrivs. Här uppstår behov av att konvertera enheterna till en och samma skala, kompatibel med mål-skalan. Ingen allmängiltig sådan skala kan anges. I vissa praktiska situationer har valts att söka uttrycka samtliga egenskaper i penningenhet: exempelvis $ eller kronor. Egenskaperna har då samtliga värderats som tillgångar i kronor (motsv), resp kostnader i kronor. En dylik teknik anses av Joslin i (19) och (20) som den bäst lämpade för databehandlingssystemutvärderingar. Vi återkommer härtill under avsnitt 6.2. Även i de fall där endast mål av typen kostnadsminimering eller vinstmaximering är aktuella förèligger vissa svårigheter vid denna konvertering. Med vilken säkerhet kan t ex ett datorsystems användarvänlighet uttryckas i kronor? I ökande utsträckning har vi att vänta att andra mål för företagen visar sig aktuella jämsides med mål av rent pekunjär natur. Moderna datorsystems flexibilitet och allmänna användbarhet talar härför. Enheten kronor skulle då knappast bli lika attraktiv, enär datorsystemen har andra viktiga karaktäristiska än vinst och kostnadsberöringar, vilka ej enkelt låter sig konverteras utan stor osäkerhet i uppskattningarna.

I de situationer där kostnadsminimering eller vinstmaximering är dominerande, och där aktuella systems omfång och flexibilitet är begränsad, är dock enheten kronor lämplig. Vi vet att dessa situationer ej är sällan förekommande. En penningenhet har dessutom den fördelen att den är begriplig för alla berörda, varför en kritisk debatt av dess användning i utvärderingsarbetet underlättas. Detta är självfallet en fördel.

Hur skall en till kronor alternativ enhet väljas? Med dess hjälp skall uppfyllelse av flera olika typer av mål kunna uttryckas. Men vi känner på förhand inte dessa måls art. Det är därför naturligt att söka definiera någon "dimensionslös" enhet för att beskriva allmän måluppfyllelse.

I sociologisk litteratur behandlas problem liknande dessa. Därvid är frågeställningen ofta omvänd, det gäller att på basis av uppmätta egenskaper (hos människor i grupp) med hjälp av lämpliga enheter konstatera vilka mål, eller vilken måluppfyllelse, som egenskaperna kan anses härröra från. Diskussioner i (bl a) Green (21), Torgerson (22) belyser detaljerat dessa och ovan berörda problem.

Skalan för måluppfyllelsen kan betraktas dels absolut och dels relativt. Önskar vi tillämpa metoder där absolut skalangivelse krävs kan bl a en skalenhet i form av poäng användas. En dylik enhet har god målflexibilitet, då den åpriori inte är relaterad till något enskilt mål.

Önskar vi använda en relativ måluppfyllelsebetraktelse kan vi implicit (och underförstått) anta existensen av en väl lämpad enhet, vars exakta natur vi emellertid därvid inte behöver känna. I dessa fall betraktar vi endast grupper (om två eller flera) av aktuella system, och anger för varje systemegenskap vilket (eller möjligen vilka) system som därvid uppvisar högst måluppfyllelse. Vi återkommer härtill.

I båda dessa fall krävs för att kunna fatta beslut om bäst lämpade system (system med högsta måluppfyllelse) en metodik för att sammansätta måluppfyllelser för de aktuella systemens enskilda egenskaper. Olika sätt att utföra detta kommer att redovisas under de nedan angivna utvärderingsmetoderna.

I tillämpningssituationer rörande systemval är det av största betydelse att systemen analyseras så att erhållna delegenskaper är oberoende av varandra. Eventuellt beroende (t ex precedensförhållande) mellan två egenskaper får den icke önskvärda effekten att den beroende egenskapen kommer att övervärderas, enär den beaktas dels ensam, och dels via sin(a) precedent(er). Ett skevt värderingsresultat erhålles därvid. Härförutom bör konstateras att det är av vikt att egenskapslistan göres fullständig. Är egenskapslistan icke

komplett har detta betydelsen att det helt bortses från icke medtagna egenskaper (vilket är ekvivalent med att dessa ges vikten noll). För diskussion härav se t ex Langefors (13), sid 21.

Till utseendet av egenskaperna för databehandlingssystem skall vi, som nämnts, senare mera detaljerat återkomma.

Det kan tilläggas att följande uppdelning av skalor (på sid 337) anges i (det mycket läsvärda) arbetet av Green (ref. (21), sid 335-366):

> "In a nominal scale objects are placed in several mutually exclusive categories, but there is no neccessary relation among the categories. The number assigned to the objects in a category serves as a label, but has no quantitative meaning.
>
> ...
>
> When objects can be arranged in a rank order with respect to some variable, they are said to form an ordinal scale. Numbers may be assigned to the objects if the numbers are in the same rank as the objects.
>
> ...
>
> When the intervals between objects on a scale can be measured, then an interval scale exists. An interval scale of hardness would require some experimental method to determine how much harder substance A was than substance B. We could then assign a number to each substance so that the difference between the two values indicated the extent of the difference in hardness.
>
> ...
>
> A ratio scale results when there is some way of showing how many times greater one object is than another. Numbers are assigned to the objects so that the numbers have the same ratio as the experimentally determined ratios of the objects."

Vår aktuella problemställning är icke helt ekvivalent med den sociologiskt betingade i (21). Vi har därför valt att i stället enbart göra uppdelningen absoluta/relativa skalor. Emellertid finns beröringspunkter. Vår poängskala kan ses som en "interval scale", och vikt-skalan (som vi nedan betraktar under rubriken "viktad poängsättning") kan ses som en "ratio scale".

6.2 Utvärdering med absolut angiven måluppfyllelse

Som ovan antytts har i många fall utvärderingar utförts med måluppfyllelsen angiven i absoluta mått. De måttenheter som främst kommit till användning är penningenhet (nedan exemplifierat med kronor) samt någon form av poäng. Nedan diskuteras metoder av dylik art. Diskussionen av varje metod kompletteras med en kritisk belysning av metodens användbarhet inklusive eventuella erfarenheter som är kända från aktuella tillämpningar.

6.2.1 "Kostnads/värde-teknik"

Denna metod, vars benämning här är direkt översatt från (19), belyses bl a i två arbeten av Joslin ref (19), sid 367 ff, samt ref (20), fr o m kap 2. Den har följande innebörd.

Utgångspunkt är att aktuellt företag arbetar med kostnadsminimering som enda mål. I ett tidigt skede av datasystemupphandlingen, helst före emottagandet av leverantörernas systemförslag, listas dels de systemegenskaper som är absolut erforderliga, och dels de som i så stor utsträckning som möjligt är önskvärda.

Efter emottagandet av leverantörernas systemförslag exkluderas de(n) leverantör(er) vars/vilkas systemförslag ej besitter de erforderliga egenskaperna.

Varje egenskap för vart och ett av de kvarvarande systemförslagen tilldelas därefter en kostnadsangivelse eller en värdeangivelse i kronor (motsv). Kostnader för egenskaper som belastar företagets resurser sammanförs därefter under en gemensam rubrik, total kostnad, samt sammanförs värden för egenskaper som är att betrakta som tillgångar för företaget (i samband med systemanvändningen) under den gemensamma rubriken totalt värde. Skillnaden mellan den totala kostnaden och det totala värdet beräknas för varje leverantör, och det systemförslag väljs som kan uppvisa den totalt lägsta nettokostnaden.

För att belysa metoden redovisas nedan ett exempel, hämtat ur (19), sid 164. I detta exempel är vissa kostnader som är gemensamma för alla de fyra aktuella systemen, ej medtagna, då de ej påverkar relationerna mellan de olika systemens kostnader.

COST-VALUE ACCOUNTING TABLE

Items	Vendor X	Vendor Y_A	Vendor Y_B	Vendor Z
Cost				
Site preparation	$ —	$ —	$ —	$ 2,000
Transportation and installation	—	—	—	—
Vendor support	—	—	—	—
Operating personnel and conversion	310,000	268,000	268,000	241,000
Equipment and maintenance	359,900	493,000	652,700	753,100
Total cost	669,900	761,000	920,700	996,100
Value				
Workload	$ 27,750	$ 95,800	$ 95,800	$128,500
Equipment expansion	—	—	75,800	125,500
Sort/merge	80,000	100,000	100,000	100,000
Automatic debug	—	—	—	60,000
PERT	—	10,000	10,000	20,000
On-site maintenance	—	40,000	40,000	40,000
Program test time	10,000	25,000	25,000	25,000
Program support	40,000	40,000	40,000	40,000
Equipment delivery	50,000	10,000	10,000	−30,000
Space available	− 7,000	—	—	—
Total value of desirable features	200,750	320,800	395,800	508,500
Cost of basic requirements	469,150	440,200	524,900	487,600

Det framgår av ovanstående att med denna metod systemet Y_A utväljs som det i det aktuella fallet bäst lämpade. Vi kan tillägga att den här redovisade metodiken nära överensstämmer med resonemang i McKean (24). Där diskuteras möjligheter att pekunjärt kvantifiera systemegenskaper av olika natur, inkluderande sådana som ur olika synpunkter är att betrakta som "intangibla". Vi kommer i detta sammanhang ej att allmänt diskutera intangibla effekter, utan hänvisar till bl a (24). Vissa egenskaper hos databehandlingssystem kan dock anses vara av intangibel natur, och till dylika egenskaper kommer vi senare att återkomma.

Synpunkter på metodens användbarhet:

1 Metoden avses ej komma till användning då andra mål än pekunjära är aktuella.

2 Joslin anger i (20), sid 16, att metoden "establishes a meaningful relationship between the(se) items of value and the system's cost". Utan tvekan är metoden fördelaktig i och med att systemens egenskaper mäts i kronor (motsv), en enhet som är gripbar för alla berörda. Dock blir med säkerhet uppskattningarna av egenskapernas värden i många fall mycket osäkra. Hur värdesätts t ex egenskaper av typen "systemutvidgbarhet" eller "flexibilitet hos operativsystem"?

3 Joslin anger i (19), sid 375: "The simplest and perhaps the best method of cost-value assignment is to simply request the other vendors to quote costs to supply a service equal to what is considered "best" in each case".

Flera skäl talar för att härmed viss osäkerhet åstadkommes, dels att andra leverantörer ofta inte över huvud kan visa upp den "bästa" systemegenskapen, och därför knappast kan kostnadssätta den, och dels att leverantörerna i realiteten oftast inte känner tillämpningsmiljön tillräckligt för att kunna ge acceptabla

kostnadsuppskattningar för egenskapers användbarhet i denna miljö. Dessutom kan sägas att även om en "annan" leverantör åtar sig att uppskatta kostnaden för att tillhandahålla en viss typ av service han ej har erbjudit, så skulle denna kostnad sannolikt i så fall avvika från det motiverade, på grund av skillnaden mellan kostnader för att introducera en service och för att löpande tillhandahålla den sedan produktionen bringats i drift (skillnaden kan ses som ställkostnaden för att starta upp produktionen av denna service, en kostnad som ej alltid kan utspridas jämnt över kommande löpande produktion).

6.2.2 Poängmetoder

Metoder där systemens egenskapers bidrag till måluppfyllelsen mäts i poäng förekommer i varianterna utan resp med viktning. Granskning av dessa situationer ges bl a i Green (21). Vi inleder redovisningen med att diskutera själva poängsättningen, som alltså är gemensam för metoderna.

6.2.2.1 Poängbegreppet och dess användning

En mångfald utvärderingar har under senare år utförts med hjälp av poängsättning av systemegenskapernas måluppfyllelse. Vi har tidigare kortfattat berör detta "dimensionslösa" begrepp, och konstaterat att det besitter den fördelen framför enheten kronor att det kan bringas att uttrycka uppfyllelse även av mål som ej är direkt vinst- eller kostndsberörande. Å andra sidan medför användandet av poäng svårigheter för av utvärderingsarbetet icke direkt berörda personer (bl a företagsledningen) att tolka systemjämförelsen, d v s att finna det informationsinnehåll som döljes bakom poängangivelserna. De fullständiga värderingsnormer som utvärderingsgruppens medlemmar arbetar med, d v s "konverteringsfunktion" från måluppfyllelse till poäng, låter sig vanligen ej enkelt dokumenteras. Detta kan vara att betrakta som en nackdel, men behöver ej nödvändigtvis vara belastande, då företagsledningen ej bör vara detaljintresserad av (kan sakna kompetens att bedöma) ett flertal tekniska systemegenskaper.

Före poängsättningen (uppskattningen av egenskapernas måluppfyllelsegrad) måste givetvis poänskalans utseende och interna betydelse fixeras. Olika skalor har här använts i praktiska situationer.De flesta kan betraktas som varianter av en s k Likert-skala (se t ex Swedner (25), sid 111), där fem alternativ används. I vår aktuella tillämpning kan vi t ex ge alternativen betydelserna:

1 poäng - mycket dålig måluppfyllelse
2 poäng - dålig måluppfyllelse
3 poäng - medelgod måluppfyllelse
4 poäng - god måluppfyllelse
5 poäng - mycket god måluppfyllelse

Denna skala är icke påfunnen på måfå. Den baserar sig på en metod som Likert lanserade år 1932, vid vilken angivna proportioner av åsikter transformerades med hjälp av en normalfördelning. Detta modifierades dock kort därefter då det upptäcktes att skalvärdena 1,2,..., 5 genererade en skala som korrelerade sig bättre än 0.99 med Likerts normalfördelningstransformationer. (Green, (21), sid 351).

Skalan 1,2,3,4,5 är en direkt motsvarighet till exempelvis skalan -2, -1, 0, 1, 2 då denna senare enbart innebär en transformation av den föregående (subtraktion med 3). Sharf anger i (26), sid 56-57, följande skala, vilken enligt honom "has been found to be most suitable".

"0.0 No value
1.0 Extremely bad
2.0 Very bad
3.0 Bad
4.0 Below average
5.0 Average, in the middle of others, medium
6.0 Above average
7.0 Good
8.0 Very good
9.0 Excellent
10.0 Perfect, Improvement without interest for us."

Beträffande antalet möjliga poäng i vald skala, d v s skalans vidd, kan framhållas:

1 Med ett stort antal möjliga poäng närmar sig poängsättningen en skattning, som grafiskt kan åskådliggöras med en rät linje, på vilken aktuell egenskaps måluppfyllelse inprickas som en punkt mellan de två ändpunkterna (se t ex Boalt-Jonsson (27), sid 97). Måluppfyllelsen kan

här mätas (mekaniskt) med ett längdmått (ex vis en linjal). Det är möjligt att algebraiskt manipulera med här erhållna längdmått på samma sätt som med poäng utan att förlora aktuellt informationsinnehåll.

3 Praktiska erfarenheter (bl a av förf) från poängvärdering talar för att ett ej alltför stort antal möjliga poäng bör användas. Med antalet fem eller sju tenderar yttervärdena att komma till användning alltför sällan. Därmed förlorar skalan i vidd vid tillämpningssituationen. Med användning av ett udda antal möjliga poäng tenderar mittvärdet att komma till användning i hög utsträckning, möjligen i högre grad än som sanningsenligt är motiverat. Det finns alltså risk att metodens validitet därmed sjunker.

4 Används ett jämnt antal möjliga poäng finns risk att ställningstaganden "provoceras" fram p g a brist på heltalig mittpunkt på skalan (delar av heltal används därvid ej). Även här kan alltså lägre validitet induceras av själva skalans utseende.

Erfarenhetsmässigt har svårigheter visat sig föreligga att kunna utföra poängsättning av en systemegenskap för olika aktuella system på ett oberoende sätt. Resonemang av typen: "Eftersom system A här bedömdes som 1 poäng, så måste system B för denna egenskap bedömas som minst 2 poäng" synes ej sällan förekomma. Systemen bör med här aktuell metodik så långt möjligt bedömas oberoende av varandra för att hög validitet i bedömningen skall uppnås. Det är vid användning av poängsättning fråga om att söka värdera varje systemegenskaps bidrag till måluppfyllelse, och ej att jämföra systemegenskaper inbördes.

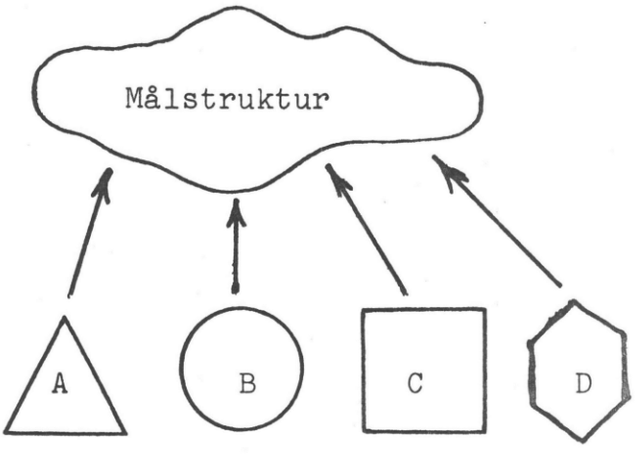


Fig 6.2.2.1:1

I enlighet med fig 6.2.2.1:1 kan vi formulera detta på det sättet att de "vertikala" relationerna söks snarare än de "horisontella".

Sharf synes i (26), sid 57, fästa mindre vikt vid nyss nämnda synpunkt, enär han beträffande poängsättningen anger:

> "Pick out the machine configuration whose element lies between the others in your qualitative/quantitative evaluation (is average) and give this machine's element 5.0 points.
> Give the elements of the other machines points in relation to this (i.e. some more than 5, some less, and some possibly the same points)".

En speciell avvägning i samband med poängmetoder rör huruvida uppenbara och konkreta kostnader, t ex av typen köp/hyreskostnader, skall bedömas som en allmän systemegenskap, och därför poängsättas m a p sitt "lönsamhetsbidrag", eller om de skall redovisas separat. Nära alla företag arbetar med någon form av kostnadsbegränsning som ett mål. - I aktuella tillämpningssituationer har ibland

valts att utdraga dylika kostnader ur poängbedömningen (enär de kan anses fullständigt säkra, och därför inte behöver "skattas"). Sharpe anger i (28), sid 285:

> "Costis seldom included as one of the factors in a scoring system",

och anser senare ((28), sid 292):

> "Assuming that an appropriate measure of performance can be obtained ... it is far better to find the system giving maximum performance for each of several levels of cost and then choose the preferred cost and performance level explicitly".

Härvid kan ett slutligt relativt prestandamått i form av prestanda/krona redovisas som basis för systemvalet. Denna enhet har dimensionen nyttovärde per penningenhet. - Sålunda har det varit mindre vanligt att poängbedöma dylika direkta kostnader, möjligen bl a p g a intrycket att poängbedömning skulle "dölja" dem, eller t o m (felaktigt) göra dem mindre säkra.

Vi kan här tillägga att poängmetoder, liksom f ö kostnads/värde-metoden, kan användas för successivt förfinade måluppfyllelseangivelser. Flexibiliteten medger vid tillämpning ett återgående för justering av ansatta värden/poäng allteftersom under utvärderingens gång ökade kunskaper och erfarenheter angående de aktuella systemen vinnes.

Det bör i anknytning till vår tidigare diskussion av målproblem framhållas att genuina multimålsituationer (se Langefors (23), sid 6) icke sammanfattat kan behandlas med hjälp av poängmetoder. I stället får uppfyllelse av varje ultimärmål för sig studeras, varpå en slutlig icke-formell bedömning görs.

Langefors gör i (23), sid 25, följande påpekande rörande jämförelse mellan skattning och poängangivelser. Han menar att

> "vi funnit anledning att ta avstånd från användning av poängbegreppet och kräver att man i stället skall använda skattningar (när ej mätning eller beräkning kan eller bör göras) av principiellt kvantifierbara effekter".
>
> .....
>
> "Verbala" värderingar likvärda med poängsättning bör reserveras för de ansvariga beslutarna och endast avse slutlig samvägning av ultimärönskemålsbidrag."

Som ett av motiven för dessa och andra påpekanden anförs dessutom i (23), sid 26:

> "T ex blir då skillnaderna mellan två alternativ med avseende på en egenskap ett relevant mått på skillnaden i nyttovärde, tvärtemot vad fallet blir med poängsättningen."

Avslutningsvis kan påpekas att få egenskaper hos databehandlingssystem (bortsett från köp/hyreskostnader) normalt är direkt och fysiskt explicit mätbara med fullständig reliabilitet. Vi återkommer emellertid senare till diskussion av dessa egenskapers utseende, och övergår härmed till att beskriva två utvärderingsmetoder där poängsättning kommer till användning.

6.2.2.2 Metoden med oviktad poängsättning

De till lämplig nivå nedbrutna systemegenskaperna bedöms med denna metod m a p bidrag till måluppfyllelse och tilldelas poäng i enlighet härmed. Under förutsättning att de aktuella egenskaperna kan anses som oberoende, kan därefter en summering av erhållna poäng utföras för varje system. (Som vi tidigare noterat får denna summering en skev innebörd om egenskaperna inte är oberoende, t ex om precedensförhållanden råder dem emellan). Det system väljs därefter som efter summeringen erhållit det högsta antalet poäng.

Metodiken är ekvivalent med konstaterande av vilket system som når störst aritmetiskt medium mellan egenskapsvärdena, enär antalet egenskaper är konstant vid systemjämförelsen.

Exemplet i fig 6.2.2.2:1 visar en kortfattad och schematisk tillämpning av denna metod, där en tregradig poängskala valts, och köp/hyreskostnader valts att ej redovisas separat, utan innefattas under posten 'övrigt'.

Egenskaper (sammanfattat)	System: Maskin A	Maskin B	Maskin C
Maskinvara	1	3	1
Programvara	3	2	2
Personvara[1)]	2	2	1
Övrigt	3	1	2
Σ	9	8	6

Fig 6.2.2.2:1 Oviktad poängsättning

1) Vi upprepar här att med personvara avses egenskaper av betydelse för kommunikation mellan människa och program.

Det framgår av ovanstående att i fig 6.2.2.2:1 maskinsystemet A väljs som det i den aktuella miljön bäst lämpade. (Självfallet är i detta exempel poängangivelserna erhållna genom summering från egenskapsangivelser på lägre nivåer än den enda i figuren angivna. Poängtilldelningen sker endast på lägsta nivån.)

Synpunkter på metodens användning:

Vi har tidigare (under 6.2.2.1) diskuterat vissa av själva poängsättningens för- och nackdelar. Härutöver kan beträffande oviktad poängsättning som utvärderingsmetod noteras att denna metod i praktiken ej kommit till användning i samma utsträckning som den viktade poängmetoden (se nedan). Detta kan möjligen ha sin orsak i att i situationer då målanalys icke tillräckligt fullständigt utförts, vägning må kunna ge de för utvärderingen ansvariga en känsla av att på ett 'nyanserat' sätt ha betraktat de olika systemens egenskaper i den aktuella miljön. Denna nyansering kan dock självfallet lika väl, om inte bättre, tillämpas utan vägning.

6.2.2.3 Metoden med viktad poängsättning

Den viktade poängsättningen tillämpar samma typer av skalor för egenskapernas bidrag till måluppfyllelse som den oviktade poängsättningen. Skillnaden mellan metoderna ligger i att vid den oviktade metoden addition sker direkt av tilldelade poängangivelser, medan i det här aktuella fallet egenskaperna tilldelas vikter före poängtilldelning och addition. Termerna för addition erhålles alltså genom multiplikation av resp poängangivelser med resp vikter. Vi kan säga att en skalärprodukt av viktvektorn och poängangivelsevektorn (nyttovärdesvektorn) bildas.

Vikterna kan väljas på olika sätt. Det kan visas (se ref i Green (21), sid 353) att en godtycklig viktad kombination är mycket högt korrelerad med vilken som helst annan viktad kombination, om korresponderande vikter har samma tecken och antalet egenskaper ej är för litet. Det framhålls även i (21) att den vid viktad poängsättning erhållna poängfördelningen når maximal varians om summan av de kvadrerade vikterna är ett. Detta har dock knappast använts i tillämpningssituationer i vårt land.

I praktiska situationer har den viktade poängsättningen tillämpats med nivåuppdelning av systemegenskaperna. Dessa har härvid nedbrutits hierarkiskt till flera nivåer (vanligen 3 - 5 st, beroende på systemomfattning), och på lägsta nivå tilldelade poäng (angivna bidrag till måluppfyllelse) har vägts och summerats uppåt i hierarkin, för att slutligt erhålla ett för hela systemet giltigt mått på måluppfyllelsen (systemets nyttovärde). Vi åskådliggör detta i figur 6.2.2.3:1.

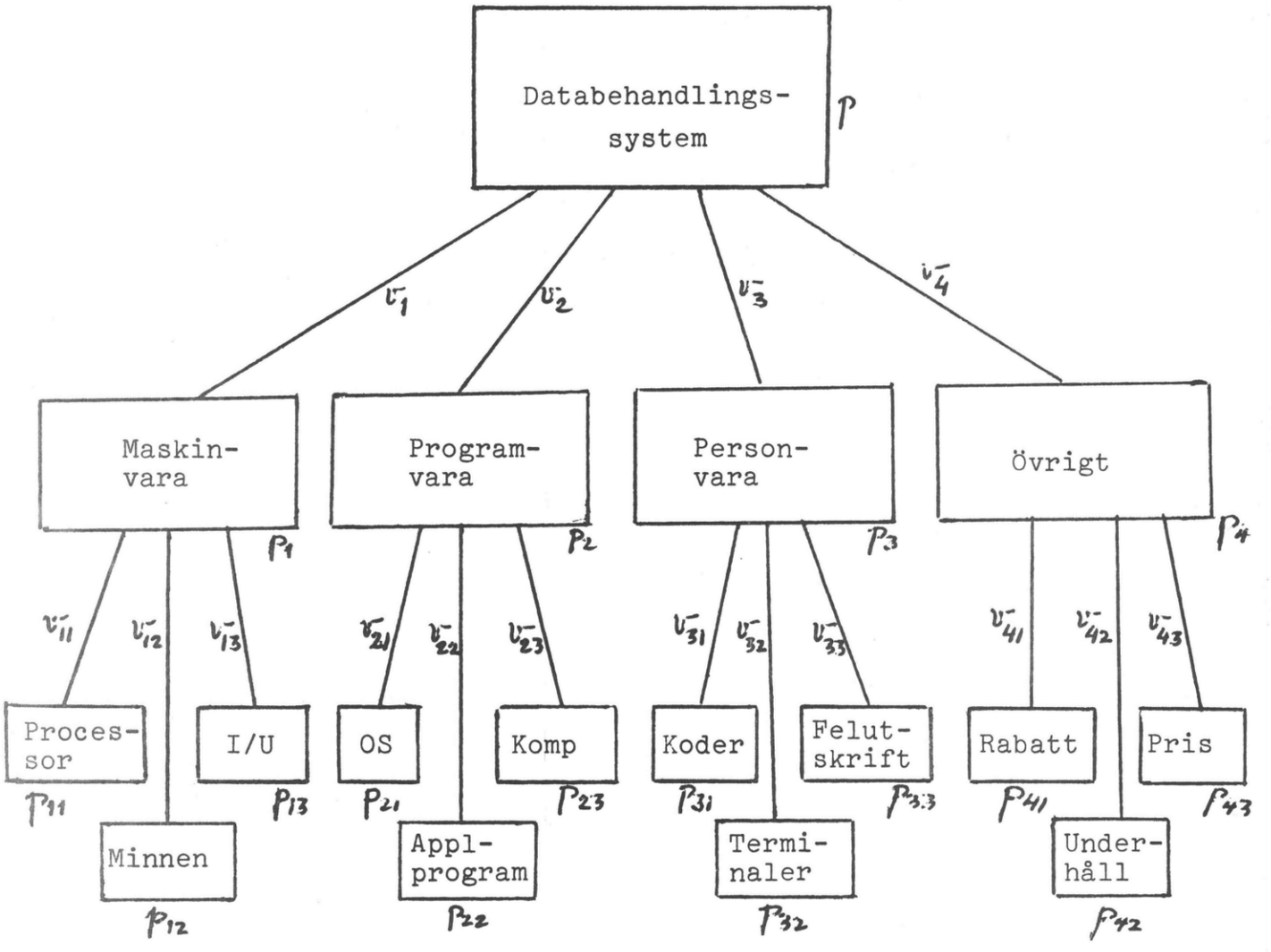


Fig 6.2.2.3:1 Viktad poängsättning (trenivåigt exempel)

I fig 6.2.2.3:1 tilldelas sålunda egenskaperna på lägsta nivå poängangivelser avseende bidrag till måluppfyllelse enligt den fastslagna poängskalan. Efter multiplikation med angivna vikter på denna nivå summeras uppåt i hierarkin, enligt följande:

$$p_k = \sum_{j \geqslant 1} v_{kj} \cdot p_{kj}$$

$$p = \sum_{j \geqslant 1} v_j \cdot p_j$$

Antalet index för p och v ovan är avhängigt hur långt nedbrytningen utförts (maxantal index = nivåantal - 1).
Vi diskuterar i detta sammanhang icke problemen vid uppdelning av databehandlingssystem i hierarkiska egenskaper. Interferenser o d mellan systemdelar komplicerar denna uppdelning, och detta belyses i ett senare avsnitt.

Metoden med viktad poängsättning har för databehandlingssystem tillämpats praktiskt i omfattande utsträckning, och flera skribenter nämner den som väl lämpad för sitt syfte. Den finns belyst (år 1964) av Rosenthal, ref (29). Senare (fr o m år 1968) har metoden tillämpats och diskuterats av bl a Scharf (se t ex ref (26), sid 53. samt ref (30)).

Synpunkter kring metodens användning:

Flera synpunkter kring ansättning av vikter må noteras. För det första bör viktsättningen utföras helt separat från poängsättningen, för att icke vikter och poäng skall påverka varandra, i vilket fall en förutsättning för metodens användbarhet skulle undanryckas. Rosenthal instämmer häri i (29), sid 364:

> "Factors (=egenskaper; förf:s anm), relative weights, and detailed instructions for the scoring, all designed for each specific selection, must be completely documented before proposals are in".

Vi uppmärksammar att Rosenthal anger att "instructions for scoring" bör fastställas före det leverantörernas systemförslag införskaffats. Detta är givetvis icke detsamma som själva poängsättningen, vilken ju icke kan utföras utan tillgängliga systemförslag.

För det andra kan vi påminna oss utsagan av Green i (21) att valet av viktskala ej är kritisk om antalet egenskaper per nivå ej är litet. Scharf använder vid tillämpning av metoden procentuella vikter på varje nivå. Han anger i (31):

> "We express the weights as a percentage of importance of the next higher level. This is a logical consequence of dividing into levels. It permits us to calculate an average value for each next higher level by simple arithmetic".

I samma ref ger Scharf därvid följande korta extraherade exempel:

"Level		Weight	Points	Weight · Points/100
3	education	40%	8.0	3.20
3	contract	20%	3.0	0.60
3	assistance	40%	4.0	1.60
2	supplier	30%	5.40	

(5,40 is weighted sum of the levels above 3.20,0.60,1.60)"

I Scharfs exempel ovan är antalet egenskaper på den betraktade nivån endast tre. Detta är ett litet antal ur metodsynpunkt. Det kan finnas skäl att överväga att välja vikter så att dessas kvadratsumma är ett, varvid resulterande angivelsefördelning får maximal varians (enligt Green (21), sid 353). Dylikt viktval blir dock något mera komplicerat att praktiskt handha.

För det tredje finner vi skäl instämma i utsagan av Scharf i (31), att vid tillämpning av den beskrivna metoden

> "a great danger ... is that we may be tempted to take into consideration the factors which are most easily

> measured, and look too much at certain factors because they seem most important for the cutcome".

Faran med att bortse från mindre lättgripbara egenskaper (av intangibel typ) delar emellertid denna metod med samtliga utvärderingsmetoder.

Joslin riktar i (19), sid 369, följande kritik mot metoden med viktad poängsättning, varvid han utgår från att explicita kostnader icke redovisas separat, utan poängvärderas liksom andra systemegenskaper:

> "The difficulty - weakness - if you prefer, in the weighted factors selection method is that points assigned to the cost section cannot be reconciled to the points assigned to the equipment characteristics section or to any other section, or even from item to item within a section".

Påpekandet av denna svårighet följs därpå av angivelsen att 'kostnads/värde-tekniken' (enligt Joslin) skulle vara undantagen detta problem. Joslin tycks därvid förbise att problemet ifråga är aktuellt för alla metoder med absolut skalangivelse för systemens måluppfyllnad, och således även för den av honom rekommenderade metoden. Problemet är icke speciellt för enbart någon enstaka skalenhet, enär databehandlingssystems prestanda och övriga egenskaper lika svårt låter sig uttryckas i kronor som i poäng eller i någon annan enhet.

Vi avslutar diskussionen av metoden med viktad poängsättning med att beakta följande påpekande av Langefors i (23), sid 7, där han i samband med analys av problemen kring viktsättning "in blanco" anger:

> "Det förhållandet att beslutarna ej i förväg kunnat eller velat väga de olika önskemålen mot varandra utesluter inte att sådan vägning kan göras vid ett aktuellt beslutstillfälle eller att det fattade beslutet i efterhand kan visas innebära sådan värdering (jfr McKean).
> Den viktiga skillnaden som görs här är mellan avvägning i förväg och in blanco å den ena sidan och mellan avvägning i samband med aktuella beslut å den andra".

I avsnitt 4 har vi diskuterat denna skillnad i synsätt och tillämpning av metoder där vägning kommer till användning. Vi har dock ansett det värt att åter fästa uppmärksamhet därpå, i samband med att den hittills mest frekvent praktiskt tillämpade utvärderingsmetoden diskuterats.

6.3 Utvärdering med relativt angiven måluppfyllelse

Hittills beskrivna metoder har angivit måluppfyllelse i absoluta mått, och olika konkreta måttskalor har därvid diskuterats. Vid angivelse att måluppfyllelse i relativa mått är de explicita måttskalornas utseende av underordnad betydelse. Härvid betraktas nämligen endast kombinationer av måluppfyllelseangivelser, och prefererade systemegenskaper noteras. För att en viss av ett systems egenskaper skall kunna prefereras före ett annat systems motsvarande egenskap m a p måluppfyllelse måste uppenbarligen någon sorts bakomliggande skala existera. Denna skalas utseende och karaktäristika är emellertid i dessa situationer av mindre intresse, enär endast själva preferenserna är informationsbärande vid denna typ av utvärdering.

Vi diskuterar nedan de olika kombinationer av preferenser som kan existera.

6.3.1 Metoden med oviktade parvisa jämförelser

Vid tillämpning av denna metod betraktas för varje systemegenskap grupper om två system, och det anges vid varje sådan jämförelse vilket system som anses kunna visa upp det högsta bidraget till måluppfyllelse. Varje angivelse betecknas som ett vunnet val.

Langefors diskuterar i (23), sid 15-17, hur precedensanalys kan användas som hjälpmedel för analys av resultat av jämförelserna (se f ö nedan). Med dylik teknik kan enkelt konstateras vilket (om något) av ett antal aktuella system som är överlägset i alla avseenden (likväl som vilket (om något) som är underlägset i alla avseenden). Enligt ett påpekande av Langefors kan hävdas att den enda vetenskapligt helt oantastliga metoden för systemutvärdering innebär val av det system som är de andra överlägset i alla avseenden (resp uteslutning av det system som är underlägset i alla avseenden). Praktiskt är dock ofta ej möjligt att

finna ett sådant system, då jämförelsevinsterna oftast är blandade mellan systemen.

Metoder med parvisa jämförelser besitter en principiell nackdel. Om antalet system som skall jämföras är stort, blir antalet par av systemegenskaper mycket stort. Med n system att jämföra måste n(n - 1)/2 jämförelser göras. (25 system, t ex, leder till 300 jämförelser). Detta begränsar i det allmänna fallet metodernas användbarhet, men bör knappast vara alltför besvärande vid datorsystemutvärderingar, eftersom antalet aktuella leverantörssystem oftast är begränsat (vanligen mindre än 10; i slutskedet av utvärderingen, då den mest noggranna studien utförs, normalt två till fyra).

Parentetiskt kan nämnas att inom sociometrin ett flertal hjälpmetoder har utarbetats för att vidga den begränsning som det stora antalet par i praktiska sammanhang ofta innebär. Hit hör "the method of rank order", "the method of equal-appearing intervals", "the method of successive intervals", m fl. Vi diskuterar i detta sammanhang icke dessa genvägsmetoder, då de enligt ovan i vårt fall knappast är erforderliga. Den specialintresserade hänvisas till t ex Torgerson (22) eller Green (21).

Varje egenskap för ett system skall sålunda jämföras med alla övriga aktuella systems motsvarande egenskap m a p bidrag till måluppfyllelse. Härvid uppenbarar sig två problem:

1 Hur skall egenskapsjämförelserna dokumenteras?
2 Hur kan en sammanfattande bild av systemjämförelserna erhållas?

Problemet 1 kan lösas på olika sätt. Vi anger två tänkbara alternativ.

a En tabell upprättas där för varje systems egenskap antalet vunna val införs. Härvid erhålles dock inge överblick över vilka val som en aktuell systemegenskap "vunnit". Detta senare kunde vara önskvärt i

samband med iterationer i arbetsgången (resultat av jämförelser kan behöva modifieras), men är i princip ej erforderligt för den sammanfattande systemjämförelsen. En dylik tabell kan få utseendet som i fig 6.3.1:1.

	System			
Egenskap	A	B	C	D
e_1	3	1	2	0
e_2	3	1	0	2
e_3	1	0	2	3
⋮	⋮	⋮	⋮	⋮
e_n	2	2	2	0

Fig 6.3.1:1 Dokumentation av vunna parvisa jämförelser

b Om vi, i likhet med Langefors i (23), sid 16, väljer att "ange det bättre av två jämförda objekt som precedent till det andra" (eller tvärtom, om vi så skulle önska), kan vi i en "egenskapsprecedensmatris" dokumentera samtliga jämförelser mellan två av systemen åt gången. För att erhålla överblick över den sammanfattande jämförelsen mellan samtliga aktuella system, får vi därvid söka överblicka ett flertal matriser (med n system, n(n - 1)/2 st matriser).

Problemet 2 ovan, beträffande sammanfattningen, kan vi med dokumentationsmetoden a lösa genom summeringar kolumnvis i jämförelsetabellen. Härvid utväljs det system som erhållit det största antalet vunna egenskapsval.

Med metoden b kan motsvarande summeringar göras, men i detta fall måste det totala antalet vunna val per system hämtas ur samtliga matriser genom rad- eller kolumnsummeringar (beroende på beteckningar), vilket får anses mer arbetskrävande.

En tänkbar variant kan innebära att kombinera de båda dokumentationsmetoderna, och alltså redovisa dels tabellsammanställning och dels aktuella jämförelseresultat i matrisform. Härvid erhålles både enkel total systemvalsöverblick och möjlighet till överblick över de enskilda jämförelserna. Detta sker dock till priset av ökat dokumentationsarbete.

6.3.2 Metoden med viktade parvisa jämförelser

Metodiken med parvisa jämförelser kan kombineras med vägning av egenskaper, som ej på lika sätt bidrar till måluppfyllelsen. Härvid kan vikter väljas utan särskilda restriktioner, eftersom endast relativa jämförelser utförs. Det är sålunda icke nödvändigt att t ex välja procentuella vikter, som för varje systems samtliga egenskaper summerar sig till ett.

Beträffande allmän tillämpbarhet av vägning hänvisar vi till tidigare (under avsnitt om de absoluta skalmetoderna) förda resonemang. Vi kan dock upprepa att vikter för de aktuella egenskaperna bör fastslås före (och alltså helt skilt från) utförandet av de parvisa jämförelserna. Tilldelning av vikter till systemegenskaperna innebär den skillnaden från oviktade jämförelser att sammanställning över de totala systemjämförelserna här innebär en summering av för varje systemegenskap antalet vunna val, multiplicerat med den tilldelade vikten för egenskapen i fråga. Sammanställningen kan i tabellform exempelvis få utseendet enligt fig 6.3.2:1.

		System					
	vikt	A		B		C	
Egenskap	v_i	n_A	$n_A \cdot v_i$	n_B	$n_B \cdot v_i$	n_C	$n_C \cdot v_i$
e_1	2	2	4	0	0	1	2
e_2	1	1	1	1	1	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
e_n	3	0	0	1	3	2	6
Σ							

Fig 6.3.2:1 Viktade parvisa jämförelser

Den under 6.3.1 nämnda sammanställningsmetoden med angivande av jämförelseresultat i matrisform kan även här komma i fråga. I stället för ettor i rutor för "vunna val" införs då resp egenskaps vikt, varvid en numerisk i stället för logisk matris erhålles. Summering kan, liksom i det tidigare fallet, ske radvis eller kolumnvis (beroende på beteckningsform) för att erhålla de prefererade av de två för jämförelse aktuella systemen. För totalöverblick över alla systemen får även här flera matriser överblickas.

Synpunkter på metodernas användning:

1 Vi kan konstatera att det synes vara naturligt för människan att arbeta med parvisa jämförelser. Det är möjligt, ja troligt, att poängsättning/skattning av bidrag till måluppfyllelse för fler än två system i själva verket baserar sig på intuitivt utförda parvisa jämförelser, och att därför denna senare systematiserade jämförelsemetod besitter fördelar i och med explicit redovisning av jämförelserna. Vi instämmer därför med Langefors i (23), sid 5:

> "Bedömning av alternativ kan blott göras för två alternativ åt gången och för dessa dessutom endast för en egenskap åt gången".

2 Ett problem föreligger avseende graderad preferens i samband med tillämpning av metoder med parvisa jämförelser. För det första må det kunna inträffa att vid jämförelse ingendera av två betraktade systemegenskaper kan föredras framför den andra. Egenskaperna kan anses helt likvärda m a p bidrag till måluppfyllelse. Härvid kan väljas att antingen ange det "oavgjorda" resultatet som "vinst" för båda, eller att lämna jämförelsen utan angivelse. Båda dessa beteckningssätt låter sig inrangeras under metodens tillämpbarhet. För det andra kan emellertid just bristen på explicit redovisad skala icke medge att stora resp små skillnader i bidrag till måluppfyllelse redovisas. Ett system kan prefereras på

basis av mycket knappt vunna jämförelser eller av "vinst med bred marginal". Skillnaden häremellan framträder ej, vilket kan begränsa den praktiska användbarheten.

6.3.3 Metoder med multipla jämförelser

Vi kan utvidga tankegångarna från de parvisa jämförelserna till att omfatta jämförelser mellan fler än två system samtidigt (för varje egenskap). Vid exempelvis triadiska jämförelser kombineras för varje egenskap grupper om tre system, och såväl "vinnare" som "näst bäst" och "förlorare" kan noteras.

Vid kvadrupla jämförelser noteras, i likhet härmed, såväl "vinnare" som "näst bäst", "näst näst bäst" och "förlorare" Vi ser att dessa metoder, f ö även metoder med parvisa jämförelser, innebär ett fastslagande av ordningsföljd för systemegenskaper m a p bidrag till måluppfyllelse, dock att ej ordningsföljden placeras in på en absolut skala. (Obs att ett ultimärmål i taget beaktas, härvid sker på expertnivå ingen målsammanslagning).

Några av de synpunkter som kan ges på metoder med multipla jämförelser innebär:

1 Självfallet måste fler system vara aktuella för jämförelse än graden av multiplicitet i metoden. Vi kan ej tillämpa kvadrupla jämförelser om blott tre olika system är aktuella för jämförelse.

2 Multipla jämförelser mellan hela system kan sammanställas och summeras i tabellform i likhet med parvisa jämförelser. Matrisrepresentation kräver emellertid fler än tvådimensionella matriser, vilka är svårhanterbara.

3 Vid multipla jämförelser erhålls en bättre överblick över graderad preferens än vid parvisa jämförelser, eftersom inte endast "vinnare" och "förlorare" noteras. Metoder att redovisa varje systems totala bidrag till

måluppfyllelse kan innebära:

a redovisning av antalet erhållna "vinnarpositioner", antalet "näst bäst-positioner" o s v tillsammans med slutlig placeringssumma.

b beräkning och redovisning av spridningen kring det summerade placeringsresultatet (eller medelvärdet därav).

4 Vägning av egenskaper kan införas, med liknande möjligheter och restriktioner som tidigare diskuterats.

5 Multipla jämförelser baserar sig på möjligheten att kunna överblicka fler än två systems egenskap åt gången. Som tidigare konstaterats är det troligt att det mänskliga naturen bäst arbetar med parvisa jämförelser, i vilket fall de multipla jämförelserna icke innebär några avgörande förbättringar i jämförelsehänseende.

6 Mer resurser krävs sannolikt, totalt sett, för att utföra multipla jämförelser än parvisa.

Vi värderar här ej dessa argument. Många andra kan tilläggas, och generella rekommendationer om metodval kan knappast ges, då metodvalet är beroende av aktuell beslutsmiljö.

6.4 Utvärdering med blandat absolut och relativt angiven måluppfyllelse

Det är möjligt att kombinera ovan diskuterade metodbaser så att såväl absolut som relativt angiven måluppfyllelse kommer till användning. För- och nackdelar för de båda betraktelsesätten följer därvid med till den erhållna produkten, och det kan knappast hävdas att nackdelar för det ena skulle kompensera (ta ut) nackdelar för det andra. Icke desto mindre kan en viss förfining av tidigare metoder möjliggöras i samband med en kombination, och vi väljer att här redovisa en dylik kombinativ metod för jämförelse mellan två system. Dessförinnan diskuterar vi en utvidgning av preferensbegreppet, som kommer till användning vid denna metod.

6.4.1 Begreppet urvalströsklar

Vid sammansättning av systems egenskapers jämförelser kan vi, antingen vi arbetar med absolut eller relativt angiven måluppfyllelse, välja att betrakta ett visst system A som överlägset ett annat, B, (m a p bidrag till måluppfyllelse) endast om A överträffar B med minst ett visst på förhand fastslaget mått p. Måttet p kan därvid ses som en undre gräns för preferens vid jämförelsen. Vid absoluta skalor får p karaktären av en absolut skalangivelse, som för tillräckligt antal egenskaper skall överträffas (ex egenskapen e_1:s bidrag till måluppfyllelse skall överstiga egenskapen e_2:s bidrag med minst x kronor, y poäng e d). Vid relativa skalor har p karaktären av ett lägsta antal vunna jämförelser (parvisa eller multipla), för att ett system skall prefereras före ett annat. Vi kallar storheten p för en urvalströskel, och noterar att den har karaktären av ett satisfieringsmål, som måste uppfyllas för preferens vid jämförelsen. Systemet A skall därvid överträffa systemet B med minst måttet p, och sålunda ej endast uppvisa "högre" bidrag till måluppfyllelse. Storheten p är därför att betrakta som ett överlägsenhetsmått.

Enär databehandlingssystem besitter en mångfald egenskaper, kan det mycket väl inträffa att systemet A överträffar systemet B med minst måttet p, men samtidigt är underlägset B i vissa avseenden. Vi kan då även definiera ett underlägsenhetsmått q, vid vilket A ej får understiga B, för preferens. Måttet q kan (liksom p) vara såväl absolut som relativt. För att preferera A före B skall därvid A överträffa B med minst måttet p och samtidigt ej vara underlägset B med mera än måttet q.

Vi har berört resonemang av "tröskelkaraktär" i avsnitt om multipla jämförelser (6.3.3), men noterar att explicit angivelse av trösklar (p, q) konkretiserar metodiken.

Urvalströsklar kan användas i samband med samtliga tidigare diskuterade metoder. Några generella utsagor om principer vid tilldelning av värden till p och q (vilket bör ske före egenskapsutvärderingen, och ej påverka densamma) kan knappast ges, enär de uppenbarligen är intimt knutna till egenskapernas natur, och må variera från fall till fall.

6.4.2 Utvärderingsmetoden Electre

Vi avslutar redovisningen av aktuella metoder för utvärdering med att diskutera en kombinationsmetod, som av sina upphovsmän givits namnet "Electre". Den finns redovisad i (32), och är diskuterad i (23), sid 18-19. Metoden, som använder såväl absolut som relativ skala för måluppfyllelse, har ingen principiellt ny innebörd, men gör bruk av begreppet urvalströsklar, och är därför av intresse.

Med denna metod jämförs två system A och B med hjälp av två index, konkordansindex c_{AB} och diskordansindex d_{AB} (s), där s är en parameter (se nedan). För beräkning av konkordansindex antas relativa vikter ha tilldelats systemegenskaperna. Med beteckningssättet att $e_K(A) \geqslant e_K(B)$ innebär att systemet A ger minst lika gott bidrag till måluppfyllelse som systemet B med avseende på egenskapen e_K definieras konkordansindex C_{AB} på följande sätt:

$$C_{AB} = \frac{\sum_{e_k(A) \geqslant e_k(B)} v_k}{\sum_{\text{alla } k} v_k}$$

Vi inser att $0 \leqslant C_{AB} \leqslant 1$. Den kommentaren kan här göras att om vikter väljs så att deras summa över alla egenskaper är 1 (exempelvis procentuellt), så försvinner nämnaren i C_{AB}. Konkordansindex gör, som vi ser, bruk av viktade parvisa jämförelse av systemens egenskaper.

Absoluta mått för bidrag till måluppfyllelse kommer till användning i samband med diskordansindex $d_{AB}(s)$. Här antas de olika egenskapernas bidrag till måluppfyllelse ha angivits i någon absolut skalenhet, varpå för varje egenskap diskrepanser b_k bildas, utgörande absoluta beloppet av skillnaden mellan egenskapernas resp bidrag till måluppfyllelse. Med $e_k(A)$ innebärande bidraget till måluppfyllelse för systemet A:s egenskap e_k har vi alltså:

$$b_k = \left| e_k(A) - e_k(B) \right|$$

Härefter kan, om så önskas, alla b_k sorteras i stigande storleksordning, för bildande av diskordansindex $d_{AB}(s)$, där vi med $b_k(s)$ avser den i storleksordning s:te diskrepansen:

$$d_{AB}(s) = \frac{b_k(s)}{\max_s (b_k(s))}$$

Vi inser att $0 \leqslant d_{AB}(s) \leqslant 1$.

För att kunna tillämpa metoden antas såväl parametern s som överlägsenhetströskeln p och underlägströskeln q ha fixerats i förväg. Metoden innebär att systemet A prefereras före systemet B om konkordansindex ej understiger p och samtidigt diskordansindex ej **överstiger** q, alltså

$$A \succcurlyeq B \text{ om } C_{AB} \geqslant p \text{ och } d_{AB}(s) \leqslant q;$$

Metoden har innebörden att systemet A prefereras före systemet B om tillräckligt många av A:s viktiga egenskaper anses ge större bidrag till måluppfyllelse än motsvarande egenskaper för B, och om samtidigt A ej är alltför underlägset B i s egenskaper.

(Vi har i denna beskrivning av Electre-metoden något modifierat beteckningarna.)

Synpunkter på metodens användbarhet:

1 Med metoden jämförs två system. Vid utvärdering av databehandlingssystem är oftast fler än två system aktuella för jämförelse. Vi har tidigare diskuterat hur härvid en slutlig sammanställning av alla parvisa jämförelser av hela system måste göras för att kunna fatta beslut om bäst lämpade system.

2 Electre-metoden gör användning av såväl vägning av egenskaper som skalärmultiplicering (summering av tvåprodukter) och parvisa jämförelser. Argument som tidigare framförts för och emot dessa förfaranden är självfallet giltiga även här. Vi upprepar dem ej ännu en gång.

3 En fördel med metoden måste anses vara att både övre och undre trösklar används. Beträffande explicit val av dessa trösklars värden kan sägas att iterationer sannolikt blir nödvändiga vid tillämpning, enär från början önskade trösklar möjligen icke alls levererar något "bästa" system (trösklarna anges initialt för optimistisk). Tröskelvärdena kan emellertid enkelt modifieras utan att egenskapsjämförelsen behöver göras om.

4 Det är givetvis möjligt att omdefiniera konkordans- och diskordansindex, om ovan angivna beräkningsnomer anses för arbetskrävande eller på annat sätt olämpliga, med behållande av tröskelprinciperna.

6.5 Utvärderingsarbetets resursförbrukning och utformning

Vi ger i detta avsnitt några korta kommentarer till utvärderingsarbetets konkreta utformning, d v s de resurser som det bör tilldelas .

1 Uppskattning av de penningsresurser som utvärderingsarbetet kräver kan belysas med hjälp av en metod som i vissa avseenden liknar den s k Mapi-metoden (redovisad i samband med litteratur om kostnads/värde-analys), och som konkretiserats av författaren. Metoden redovisas i appendix 1.

2 Utvärderingsarbetets omfång bör i största möjliga utsträckning fixeras på förhand. Här avses antalet till detta arbete tilldelade manmånader. Ett avvägande berör hur många personer på expertnivå som bör deltaga i utvärderingen.

   Antag att en utvärdering för ett databehandlingssystem som är avsett att ungefärligt befinna sig i 10Mkr-klassen (köpepris), tilldelas ca 50 manmånaders utredningsarbete. Med en utvärderingsgrupp om 10 personer skulle arbetet då (möjligen) kunna klaras av på 5 månader, och med 25 personer i gruppen skulle 2 månader räcka. Detta är ett i praktiken orealistiskt resonemang, även om vi skulle söka bortse från Parkinsons lag. Det finns (nästan alltid) en undre gräns för arbetets tidsutsträckning, som ej kan underskridas. För det i exemplet aktuella systemet är det erfarenhetsmässigt sett olämpligt att överväga ett utvärderingsarbete på kortare tid än ca 6 månader. Flera i arbetet ingående moment kan inte pressas in under kortare tidsrymd. Vi kommer att diskutera dessa moment i ett senare avsnitt.

Under 1960-talet har i praktiken ungefärligt den tumregeln kommit att gälla, att utvärderingsarbetets tidsutsträck-

ning i månader varit nära lika med (eller i varje fall ej understigit) det avsedda köpepriset i Mkr, oberoende av hur många personer som deltagit i detta arbete. Regeln är givetvis mycket grov. Då priser på databehandlingsutrustning under 1970-talet sjunker, och systemens flexibilitet ökar, blir tumregeln ej längre värdefull, eftersom den undre gränsen för arbetets utsträckning blir för grov.

Det kan anses lämpligt att i utvärderingsgruppen icke samla alltför många personer. Erfarenheter talar för att kommunikationssvårigheterna inom en expertgrupp om ca 10 personer ofta blir betydande. Det är därför klokt att till själva utvärderingen tilldela, säg, 3 - 6 personer, beroende på storlek och flexibilitet hos det avsedda systemet.

Vi nämner i dessa resonemang då och då begreppet "pris, storlek och omfång för det avsedda databehandlingssystemet". Dessa parametrar är givetvis icke kända med nämnvärd exakthet före utvärderingen. Trots detta bör i många fall grova ansättningar av dessa parametervärden kunna göras före utvärderingsarbetets start. Vi kommer att diskutera dessa spörsmål i avsnitt 7.

Vi kan i detta sammanhang notera en kommentar av Joslin i (19), sid 369, där han diskuterar ofta betydande personliga fluktutationer avseende poängsättning inom en utvärderingsgrupp, vilket han betraktar som en avsevärd nackdel för metoden med viktad poängsättning:

> "So subjective is this system that "X" number of evalutors, if requested to assign point scores to a list of elements considered important to a computer selection, would produce "X" number of lists; each selective, each the best from the evaluator's point of view, and each different".

Joslin tycks här, vilket f ö framgår på andra håll i (19), utgå från att en objektiv syn på utvärderingen skulle

existera, och att varje personlig avvikelse från denna är olämplig. Han använder sålunda ofta begreppet "objektiva bedömningar". - Vår ståndpunkt är den rakt motsatta: ingen "objektivitet" kan förväntas noteras inom en utvärderingsgrupp av normalt utseende. Gruppens arbetsresultat baserar sig helt på de i gruppen ingående individernas personliga ståndpunkter.

I utvärderingsarbetet bör en sådan arbetsform väljas att gruppmedlemmarna ges bästa möjlighet att uttrycka egna åsikter om systemprestanda o d. Detta innebär att det ofta kan vara lämpligt att bedömningar först utförs av individerna separat (i enrum), och först därefter diskussion av ståndpunkterna vidtar. Gruppens slutbedömning blir ett lämpligt medelvärde av de individuella ståndpunkterna, eventuellt modifierade på basis av löpande informationsutbyte.

För att nå fram till slutlig ståndpunkt kan i besvärliga men viktiga beslutssituationer exempelvis Delphiteknik komma till användning.

6.6 Slutsatser ur redovisade utvärderingsmetoder

Vi avslutar härmed vår diskussion av olika metoder för principiell utvärdering av databehandlingssystem. Vi har kunnat konstatera att metodernas användning icke begränsar sig till just dessa typer av system, utan är användbara i många andra sammanhang. Detta följer av att vi icke har behövt specialisera oss på databehandlingstekniska resonemang (annat än i vissa undantagsfall).

Det kan vara på sin plats att här återblicka på den i avsnitt 1 angivna arbetshypotesen angående databehandlingssystems miljöberoende. Har material framkommit som kan användas för verifiering eller deverifiering av denna hypotes?

I avsnitt 4 fann vi att kategorisering av mål till olika typer är miljöberoende. Detta påverkar positivt styrkan i vår hypotes.

Ur konstaterandet att de i avsnitt 6 redovisade utvärderingsmetoderna kan anses generella, och således icke enbart giltiga för databehandlingssystem, samt att vi därför icke koncentrerat oss på databehandlingstekniska aspekter vid diskussionen, följer att anknytningen till den aktuella hypotesen, där databehandlingstekniska värderingar berörs, endast är måttlig hittills. Någon fast koppling till hypotesen har vi i avsnitt 6 ej kunnat notera.

Det framgår emellertid att inga argument hittills framkommit, som motsäger den aktuella hypotesen.

Vi stannar här upp med detta konstaterande, i vetskap om att hypotesen mera ingående kommer att beröras av den fortsatta diskussionen om det konkreta utseende av de egenskaper som utmärker databehandlingssystem.

7 Egenskaper för databehandlingssystem

Vi avser nu diskutera de typer av egenskaper som kännetecknar databehandlingssystem. I resonemangen kommer vi (så långt möjligt) att undvika att enligt någon norm värdera dessa egenskaper. Värdering kan endast göras i relation till aktuella mål, och vi ansätter inga dylika för den kommande diskussionen. Avsikten är i stället att fästa uppmärksamhet på systemegenskaper i sig, vilka i tillämpningssituationer senare kan betraktas och eventuellt värderas i sin avsedda miljö.

För att diskussionen i så stor utsträckning som möjligt skall kunna bli av värde för tillämpningar kommer vi icke att begränsa oss till enstaka mer eller mindre distinkta typer av databehandlingssystem. I stället vill vi hålla resonemangen på en allmän nivå, sålunda obundna av i vilken utsträckning system idag existerar som besitter diskuterade egenskaper. Problemet att i det enskilda fallet utvälja bäst lämpade systemegenskaper får således här endast en principiell belysning. Något explicit svar på frågor av arten: "vilken typ av minidatorsystem i prisklassen 1 Mkr är bäst lämpat för tekniskt-vetenskaplig databehandling i time-sharingmiljö?" kommer alltså ej att framträda. Vi betraktar här användarmiljön som icke varierande, och beaktar icke det eventuella dynamiska samspelet mellan förändringar rörande miljö och datorsystem. Vi konstaterar endast att olika miljöer existerar, samt att dessa ställer olika krav på systemen.

I analogi med systemarbetsprinciper av Langefors i (6) betraktar vi först systemegenskaper på en mycket grov nivå, för att senare successivt spjälka upp egenskaperna i så långt möjligt distinkta finare delar.

7.1 Systemstorlekar

Vi ställer här frågan om det existerar normer, enligt vilka databehandlingssystem kan grupperas och klassificeras på ett avskiljbart sätt. Vad vi eftersträvar är en eventuell norm för beskrivning av systemens användbarhet. I utvärderingssammanhang är vi mindre intresserade av inre systemegenskaper än yttre, vilket bör ha framgått av vår tidigare diskussion.

Inledningsvis konstaterar vi att det från vissa håll är populärt att klassa ett databehandlingssystem som "första", "andra" eller "tredje" generationens system. Även en "fjärde" generation har stundom börjat användas. Denna indelning baserar sig ungefär på följande:

första generationen	- datorer med radiorör som väsentliga byggelement
andra generationen	- datorer med transistorer som väsentliga byggelement
tredje generationen	- datorer med integrerade kretsar som väsentliga byggelement
fjärde generationen	- (oftast) mera höggradigt parallellt användbara (reelltids)system

Detta sätt att klassa databehandlingssystem har bl a visat sig journalistiskt lämpligt. I mera seriösa miljöer används emellertid generationsbegreppet knappast. Det är uppenbart att den tekniska karaktären av de inre maskinvarumässiga byggstenar som ett databehandlingssystem är uppbyggt av icke väsentligen kan kännetecknа systemets uppträdande utåt. Eftersom detta uppträdande, d v s systemets användbarhet, primärt intresserar oss i utvärderingssammanhang, finner vi det olämpligt att använda generationsbegreppet. För övrigt lägger

vi märke till att den fjärde generationen definierats diffust. Den har i verkligheten tillkommit som en ren tidsmässig extrapolation från de tidigare. Den fjärde och eventuella senare generationer låter sig knappast på ett lämpligt sätt klassificeras utgående från de tre första, emedan integrerade kretselement, ofta benämnt LSI (large scale integration), är ett så allmänt och accepterat byggbegrepp att det sannolikt ej kommer att lämnas på lång tid. Även den ultimärt tänkbara (men osannolika) tekniken med kompletta datorer som hela gjutna halvledarblock kan i princip anses ingå under den tredje generationen.

I detta sammanhang kan vi påminna oss följande uttalande av J P Eckert (ca år 1970):

> "I know of no significant new computer technology achievment since 1960".

Vi finner skäl att ej vidare diskutera generationsbegreppet i samband med eventuell klassificering av datorsystem.

Viktigare är att söka utreda om "storlekar" av datorsystem existerar, med gruppvis liknande egenskaper. Om så är fallet blir situationen enklare hanterbar för företag som söker tänkbara datorsystem för sina informationssystem. I bl a offertsammanhang skulle existensen av ett väl definierat storleksmått väsentligt underlätta kontakterna mellan kund och leverantör, emedan därvid referenser löpande kunde göras till detta mått vid diskussion av databehandlingsbehov.

Existerar det sålunda ett (eller flera) egenskapsmått för att uttrycka ett datorsystems "storlek"? Uppenbart är att vi närmare behöver precisera själva storleksbegreppet för att kunna besvara frågan.

Databehandlingssystem har till uppgift att "behandla" data. Detta kan innebära alla tänkbara typer av behandling, såsom t ex grammatikaliska, logiska, matematiska

manipulationer m m. Utan att här närmare gå in på dessa typer kan vi konstatera att snabbheten med vilken manipulationerna utförs (prestationsförmågan) är av intresse i alla sammanhang där tiden är intressant som en variabel eller parameter i målstrukturen. Vi utgår från att detta är fallet i de flesta miljöer, och finner alltså datorsystems snabbhet principiellt intressant. (Självfallet är detta ej den enda intressanta egenskapen).

Emedan icke alla miljöer ställer liknande krav på datorsystemen (vilket vi tidigare diskuterat) uppkommer härefter frågan om något generellt snabbhetsmått existerar, eller om snabbhet endast kan betraktas med avseende på vad som behandlas. Den senare delen av detta spörsmål, att snabbhet avseende en viss (eller vissa) tillämpning(ar) är intressant finner vi så naturligt att vi för tillfället lämnar densamma. Vi återkommer emellertid senare i denna fråga.

7.1.1 Grosch's lag

Existerar något generellt snabbhetsmått för datorer? En stor mängd författare och forskare har ägnat sig åt detta problem. Merparten av dessa har startat diskussionen med den s k Grosch's lag. Denna lag, som formulerades av H. Grosch vid slutet av 1940-talet (exakt tidpunkt okänd), utsäger att ett datorsystems snabbhet på att lösa sina uppgifter (stundom expanderat till att kallas prestationsförmåga) är direkt proportionellt mot kvadraten på systemets kostnad. Vi kan också formulera den så att kostnaden är direkt proportionell mot kvadratroten ur prestationsförmågan.

$$C = K \cdot \sqrt{E}$$

$$E = \frac{1}{K^2} \cdot C^2$$

där

C = systemets kostnad
E = systemets prestationsförmåga, effektivitet
K = en konstant

(för referens se t ex Sharpe (28), sid 315).

Vi lägger märke till att denna lag endast ger en relativ utsaga om datorsystems prestationsförmåga. Med kännedom om två aktuella systems kostnader (C_1 och C_2) samt det enas prestationsförmåga (E_1) kan det andras prestationsförmåga (E_2) anges:

$$C_1 = K \cdot \sqrt{E_1}$$

$$C_2 = K \cdot \sqrt{E_2}$$

$$E_2 = \frac{C_2^2}{C_1^2} \cdot E_1$$

Ett flertal studier har utförts för att utröna för vilken typ (vilka typer) av prestationer som lagen är giltig, och i så fall med vilken precision. Att perfektion ingalunda gäller generellt, för alla tänkbara tillämpningar, har de flesta tagit som utgångspunkt för diskussion av lagen.

Innan vi fortsätter studierna av denna lag, och de meningsutbyten angående effektivitet för olika maskinstorlekar som den givit upphov till, vill vi nämna något om ytterligare försök till "generella" mått på datorsystems kapacitet eller prestationsförmåga.

7.1.2 Andra effektivitetsmått

Relativt få försök att finna enkla analytiska måttenheter för datorkapacitet har gjorts, medan de empiriska undersökningarna är desto fler. Vi nämner här några lanserade analytiska och absoluta (i motsats till relativa) enheter. Dessa konstaterar det uppenbara att samspelet mellan processorkapacitet och primärminnesstorlek har betydelse för prestationsförmåga i stort.

Schneidewind (1966) samt Skattum (1967), ref i Sharpe (28), sid 297, samt (dem oberoende) författaren till detta arbete (1968), har diskuterat följande effektivitetsmått:

$$E = \frac{M}{t}$$

där E = effektivitet
M = primärminnesstorlek
t = cykeltid

En nackdel som följer med användningen av primärminnesstorleken (M), mätt i ord, bytes eller bitar, i denna (och även nedanstående) måttenhet(er) uppkommer p g a att det på senare tid (fr o m slutet av 1960-talet) visat sig bli allt vanligare med maskinvarumässigt uppdelade primärminnen. Vanligt är sålunda att mindre, mycket snabba minnesområden (några hundra macrosekunders cykeltid och ca 10-100 ord) kompletterar det större primärminnet. Härvid kan med effektiv programmering (på assemblynivå) mycket snabba program erhållas, medan från problemorienterade språk erhållna program ofta inte förmår utnyttja minnesuppdelningen effektivt. I alla händelser medför en dylik, och ofta än mer uppdelad, primärminnesorganisation svårigheter att finna ett lämpligt och representativt värde på M.

I ref (33), sid 79, har ett försök gjorts att klassa "stora" datorsystem enligt ett mått som nära överensstämmer med Schneidewind/Skattum's. Skillnaden består

i att cykeltiden utbytts mot additionstid (i nämnaren). De vid varje tidpunkt mest kraftfulla maskinerna tycks därvid tämligen väl följa en rät regressionslinje, vilket framgår av följande diagram (ur ref (33)):

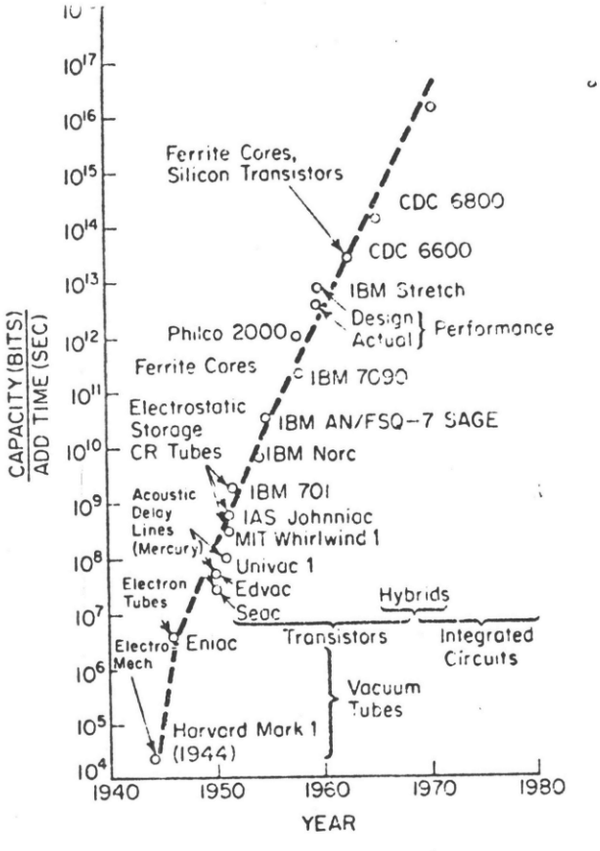


Fig 7.1.2:1 Trend för stora datorsystem

Problemen kring ansättning av lämpliga värden på M diskuteras emellertid ej i ref (33), vilket får anses olyckligt.

En annan enhet, som lanserats av Gruenberger (ref i Sharpe (28), sid 298), utbyter cykeltiden mot ett något mer komplicerat uttryck:

$$E = \frac{M}{L} \cdot (N_a + N_m)$$

där E = effektivitet
M = primärminnesstorlek (i bitar)
L = instruktionslängd (i bitar)
N_a = antalet additioner per sekund
N_m = antalet multiplikationer per sekund

Denna enhet skiljer sig i huvudsak från den tidigare givna i det att instruktionslängden (L) inkluderats. Som skäl härför anger Gruenberger i samma ref:

> "L attempts to measure inefficiences due, for example, to decimal capability"

Emellertid anges ej om L avser "medelinstruktionslängd" i det fall variabel instruktionslängd används. En heller preciseras om N_a och N_m avser fix eller flytande aritmetik, decimalt eller binärt. Gruenberger konstaterar själv (i samma ref):

> "None of the ... factors is wholly objective ... and some are extremely difficult even to estimate for some machines".

Denna halva reträtt syns representativ för flertalet av de försök som gjorts att finna analytiska, <u>generellt</u> giltiga, måttenheter för prestanda.

Ytterligare en måttenhet har lanserats, av Mathews i (34), sid 24. Enheten har utseendet:

$$E = \frac{S \cdot a}{b}$$

där E = effektivitetsmått
S = månadshyra ($)
a = additionstid (mikrosek)
b = ordstorlek (bitar)

Ju effektivare maskin, desto lägre E-angivelse, noterar Mathews. Vi återkommer till den möjliga relevansen i användning av köpepris/månadshyra i dylika enheter i samband med den vidare diskussionen av Grosch's lag.

Avslutningsvis vill vi nämna ett index som huvudsakligen anges mäta primärminneskapacitet, och som lanserats av C.J. Shaw, ref i Weinwurm (35), sid 209. Detta index har utseendet:

$$\text{Shaw index} = \frac{K \cdot b}{t} \cdot {}^{2}\log(b \cdot M)$$

där $K = 3.6 \cdot 10^{9}$

b== ordstorlek (bitar)

t = cykeltid

M = primärminnesstorlek (ord)

För "vanliga" konfigurationer av några förekommande datorsystem ger detta index följande klassificering:

System	Shaw index
IBM 1401	0.031
IBM 1410	0.086
IBM 7044	1.044
IBM 7090	1.186
IBM 7094 II	1.864
IBM 360/30	0.267
CD 1604	0.740
CD 3600	2.959
GE 235	0.207
GE 425	0.433
PDP 1	0.209
Univac 1107	0.686

fig 7.1.2:2 Klassificering med Shaw index
(ur Weinwurm (35), sid 210)

Detta index har använts för studier av programmerings-effektivitet för olika typer av program på olika dator-system. Vi återkommer till dylika empiriska undersök-ningar.

Begränsad diskussion av vetenskaplig karaktär har sålunda hittills koncentrerat sig på enheter liknande de ovan nämnda.

Detta faktum kan vi, dock hittills med otillräcklig relevans, tänka oss som belägg för den möjliga korrekt-heten i vår inledande hypotes (om prestandas miljöbero-ende). Ty hade prestanda med rimlig säkerhet kunnat anges i absoluta mått, hade åtminstone vetenskaplig diskussion härom varit märkbar i nämnvärd omfattning.

7.1.3 Storlek och effektivitet

Vid varje diskussion av "economy of scale" är det självfallet nödvändigt att definiera ingående begrepp. I samtliga utförda undersökningar, som författaren har kännedom om, har effektivitet sökt mätas i termer av typen "throughput", "performance", "capacity" e. d. Sharpe konstaterar i (28), sid 296:

> "The measure may be the number of hours required to perform some specified set of tasks or the number of such set of tasks that can be performed in a specified time period. The goal is usually to measure performance for a "typical" set of tasks. The importance of selecting an appropriate set cannot be minimized".

Trots dessa uppenbara förhållanden förekommer i litteraturen ofta generaliserade uttalanden om relationerna mellan storlek och effektivitet. Intressant är att dessa ingalunda alltid står i överensstämmelse med varandra. Vi ger här två exempel. I (36) framhåller Orlicky på sid 87:

> "The fact is widely known that it is cheaper to use a bigger rather than a smaller computer - cheaper per job processed. It costs less to execute an instruction, to add two numbers, to store a character in memory or to keep a record in random-access file with a large computer system. This phenomenon is not some marketing tactic of computer manufacturers but it is a natural consequence of computer design and structure."

Å andra sidan framhäver Mathews i (34), sid 25 följande:

> "... the fact that certain general trends are clearly evident; the most important of the trends is this: the cost per computation is becoming higher for big machines than for small ones."

För att kunna bringa samordning mellan dylika diametrala utsagor (om nu detta är möjligt) är det naturligt att undersöka om utsagorna verkligen bygger på liknande definitionsmaterial. Vi har tidigare konstaterat att enhetliga analytiska beskrivningsnormer för "storlekar" på datorsystem, som är allmänt accepterade, knappast kan sägas existera. Vilken konceptuell bakgrund bygger då uttalanden som de två ovan nämnda på?

Det förefaller som om begreppet minidator (för närvarande) till sin innebörd icke rymmer alltför omfattande osäkerhet. Variationerna är visserligen knappast små vad avser gränsdragning mot andra grupper av datorsystem, men dessa variationer motverkar dock ej en någotsånär allmän gemensamt förståelse för begreppet. Följande uttalande, av C.Schneider i (37), sid 26, kan sägas vara accepterat av stora grupper:

> "Wenn die Abgrenzung des Kleincomputers nach unten (Abrechnungsmaschinen) wie nach oben (EDVA) (= Eletronischen Datenverarbeitung-Anlagen; förf:s anm.) nicht eindentig ist, so ist aber offenbar, dass die Kleinrechenanlagen nur bei begrenzter Aufgabenstellung einsetzbar sind, während die Standard-Computer ... universelle Anwendung finden können".

Detta resonemang, att söka definiera - eller åtminstone avgränsa - datorsystem med hjälp av den användning de är lämpade för, syns av största betydelse. Vi kan formulera om utsagan till: minidatorer är främst avsedda för dedikerade tillämpningar.

Å andra sidan måste vi notera att minidatorernas möjliga användningsmiljö successivt expanderas, mindre så att de blir mer och mer generellt användbara (varvid de enligt ovan skulle upphöra att benämnas minidatorer) än så att den miljö expanderas, inom vilken dedikerade tillämpningar för minidatorer förefinns. R.A.Zack formulerar i (38), sid 36, följande:

"The variety of applications will continue to broaden as minicomputers become smaller and less expensive, a trend that has shown rapid momentum, despite the severe inflation being experienced in most other economic sectors".

Koncentrationen på datorsystems användningsmiljö ger oss anledning undersöka om gemensamma kännetecken kan sägas existera för de miljöer, i vilka olika typer av datorsystem är olika väl lämpade. Vi har nämnt begreppen "dedikerade" system i motsats till "generellt användbara" system. Med den flexibilitet som moderna datorsystem medger kan vi knappast entydigt pricka in de explicita tillämpningar som t ex minidatorer är lämpade för, men vi kan något beröra de allmänna egenskaper, som kännetecknar resp tillämpningar. Anknytande till moderna trender att beteckna minnet som det mest fundamentala elementet i ett datorsystem kan vi säga att

minidatorsystem kännetecknas av att de är lämpade för bearbetningar med endast måttliga behov av access till stora, varierande datamängder.

För större, generellt användbara system är kännetecknen de omvända. I fullständig överensstämmelse med ovanstående erfarenhetsmässigt belagda konstaterande kan vi citera E.C. Berkely i (39), sid 7, där han gör följande uppdelning av tillämpningar:

- "Problems which may be solved with very little access to large continually changing files of information, such as nearly all computation."
- "Problems that require a great deal of such access, such as making airline reservations, where failure to have access to the central file at any desired moment is a fatal defect."

Vi kan notera att inte bara exekvering av tillämpningsprogram utan även programmering kan inrangeras under ovanstående indelning. Avancerade och generellt användbara språk, som Simula 67, Algol 68 och PL/I, leder till omfångsrika kompilatorer, som samarbetar med stora filer. Detta gäller oberoende av om systemen arbetar i satsvis eller tidsdelningsmiljö. Clapp anger i (40), sid 15, à propos den visserligen ej ringa, men dock begränsade, flexibiliteten för minidatorer jämfört med (terminaler till) större system:

> "However, the small computer by its very nature is limited in size and flexibility. Because of these limitations it is very difficult to reproduce all of the features of a truly sophisticated interactive time-sharing language on a small computer."

Som en sammanfattning av resonemangen i detta avsnitt kan vi konstatera att argumenten för och emot decentralisering av datorresurser bör kunna förenas genom begreppsmässig koncentration på systemens användningsmiljö. Vissa interna resurser i ett datorsystem må kunna utnyttjas effektivare i större system än i mindre, men för användning av systemen är många andra faktorer än t ex internt optimerat processorutnyttjande av intresse.

Det är sålunda intressant att se fram emot tillkomsten av datornät, där både decentraliseringens och centraliseringens fördelar bör kunna förenas, utgående från användningsmiljöns önskemål på systemen. En omkopplingsbarhet mellan olika systemresurser i ett nät av såväl stora som små datorer bör medge bästa flexibilitet. Avslutningsvis kan vi därför instämma med de Greene i (41), sid 292-93:

"... the increasing availability of small inexpensive reliable computers will reduce the need of many users to share time and cost. Such users could perhaps "hook up" to a network when they needed help in solving problems beyond the capability of their small computer or when they required access to large files of data."

7.1.4 Effektivitet i användarmiljön

I våra diskussioner av effektivitet för olika typer ("storlekar") av datorer har vi funnit det önskvärt att skilja mellan "intern" och "extern" effektivitet i utnyttjandet av systemen. Det är höggradigt plausibelt att denna skillnad är basen för så skilda uttalanden för resp emot decentralisering av datorresurser, som givits ovan (i avsnitt 7.1.3).

Det förefaller sålunda naturligt att intern effektivitet talar för en viss centralisering av resurserna, medan den externa effektiviteten befrämjas av decentraliserade datorresurser.

Vi har tidigare, på ett allmänt plan (i kap 2), gjort ett uttalande med denna substans: alltefter som det system vidgas, inom vilket en resurs kommer att utnyttjas mindre internt effektivt. Vi avser då att samspelet med övriga resurser inom systemet får allt större betydelse.

Dessa tankegångar får stöd i följande uttalande av Amdahl i (42), sid 67:

> "On the one hand, economy of scale for processors and file devices tends to favor centralization. A strong system programming capability also favors centralization. On the other hand, there is typically a pronounced desire for local processing capability. The local manager has better control of priorities in a decentralized system, has greater confidence in file protection procedures, and feels less susceptible to central system crashes and data communication failures".
>
> ...
>
> "The effectiveness of a given system can be expected to be strongly flavored by the degree of corporate decentralization that is involved".

I de användarmiljöer, där effektivitet i ett större sammanhang (än internt datatekniskt) bedöms värdefullt, är alltså sannolikt en alltför hård centralisering mindre lycklig. Detta förhållande bör vara giltigt för stora grupper användare av databehandlingssystem. Utgående härifrån formulerar Langefors i (43), sid 6:

> "... computer selection will be based on convenience and quality of service to the user rather than on what might be saved in computer cost, by size for instance."

7.1.5 Undersökningar kring Grosch's lag

Som vi tidigare angivit är Grosch's lag den utgångspunkt som de flesta prestandaanalytiker tagit för sina arbeten. Inget av de övriga ovan nämnda klassificeringsmåtten har rönt tillnärmelsevis lika stort intresse. För att kunna ta ställning i frågan rörande möjligheter att klassindela databehandlingssystem, i samband med våra studier av dessa systems egenskaper, är det därför önskvärt att närmare belysa utförda studier av Grosch's lag.

W.J. Erikson accepterar i Weinwurm (35), sid 121-22, denna lag som intressant, men anger:

> "There is much controversy regarding the validity of this law, but it seems to be supported by analysis of data (ref), and no more useful relationships have been proposed".

En precisering för lagens möjliga giltighet görs av Almdahl i (42), sid 67, i samband med diskussion av system med multipla processorer:

> "Systems with multiple processors have developed rather slowly. Perhaps this is what should have been expected since multiprocessors are blatant violations of Grosch's law. For example, consider a configuration of two indentical processors. Grosch's law indicates that for twice the cost one should expect four times the performance. But for twice the processor cost one will find he has considerably less than twice the performance in dual processor configurations".

Till detta kan sägas att vid tiden för tillkomsten av lagen (slutet av 1940-talet) inga multiprocessorsystem existerade, eller troligen ens var påtänkta. Grosch avsåg uppenbarligen (förhållandevis) hela system (dock med tonvikt på processorerna), och ej enbart delar av system.

I annat fall skulle t ex dubblering av ett sekundärminne, en kortläsare, e d kunna väntas fyrdubbla prestanda avseende resp enhet, ett förhållande som faller på sin egen orimlighet. I begreppet "hela system", som i och för sig måste anses flexibelt, bör vi möjligen inkludera programvaror. Däremot är det tveksamt om personvaror inkluderades av Grosch. Den verkliga betydelsen häri torde knappast ha stått klar vid slutet av 1940-talet.

En empirisk studie av den möjliga relevansen i tankegångar liknande Grosch's lag har utförts av Solomon i (44). Materialet för studien valdes därvid på så sätt att osäkerheten i kostnadsangivelse för de aktuella maskinsystemen kan anses låg. Detta är fallet på grund av att de system som jämfördes var de vid nära samma tidpunkt (år 1964) introducerade IBM-systemen 360/30,40,50,65 och 75. Det måste anses rekommendabelt att, som Solomon, välja system som i så liten grad som möjligt prismässigt är under påverkan av olika kostnadskonjunkturer. Osäkerheten i jämförelserna blir annars stor, i de sammanhang där systemkostnaden spelar en väsentlig roll för jämförelserna.
I Solomons fall kunde hävdas att skillnader i förhållandet mellan kostnad och effektivitet i huvudsak borde bero på skaleffektivitet ("storlek"), och ej i nämnvärd utsträckning på marknadsföring, teknologisk utveckling etc.

För att söka ge rättvisa åt de olika systemen valde Solomon att precisera s k "average systems", vad gäller konfiguration med minne, yttre enheter m m. Han framhåller härvid:

> "When comparing small machines with large ones, the large computers must neccesarily be complemented with more devices or else the economies of scale are meaningless".

Kenneth Knight har en motsvarande studie (ref (49), sid 31 ff) angivit vad han anser vara "rimliga" maskinkonfigurationer. Sharpe diskuterar i (28), sid 317, dessa "genomsnittliga" system, och sammanställer kostnaderna för

systemen, relaterat till 360/75 som 1.000:

Model	Solomon's genomsnittliga system	Knight's typiska system
360/75	1.000	1.000
360/65	0.625	0.852
360/50	0.400	0.430
360/40	0.213	0.218
360/30	0.100	0.162

Figur 7.1.5:1 Typiska relativa kostnader för IBM-system

Vi ser att skillnaderna, möjligen bortsett från 360/65, endast är måttliga.

Solomon fann det befogat att precisera meningen med begreppet "effektivitet". För att nå relevans i mätningarna lät han systemeffektiviteten speglas i förmåga att bearbeta vissa givna uppgifter. Dessa bearbetningar utgjordes av

1 ett problem med dominans för matrismultiplikationer ("highly scientific")

2 ett problem med flytande kvadratrotsberäkning ("also scientific but utilizes arithmetic capabilities less heavily")

3 "Field scan of a card for control options" ("perhaps more closely related to data processing (and compilation) applications")

4 en "vetenskaplig" instruktionsblandning enligt Arbuckle i (45).

Dessa problems instruktionskaraktär uppdelades (grovt) m hj a vikterna:

	Problem, typ		
	1	2	3
Fixpunktsoperationer	0.3682	0.0634	0.2704
Flytande operationer	0.4136	0.6349	0
Logiska operationer	0.0002	0	0.2260
Hopp	0.2178	0.2858	0.5021
Status hantering	0.0002	0.0159	0.0015
	1.0000	1.0000	1.0000

Problem, typ 4 (Arbuckle):

Flytande add/subtr	0.095
Flytande mult	0.056
Flytande div	0.020
Laddn/lagring	0.285
Indexering	0.225
Villkorliga hopp	0.132
Diverse	0.187
	1.000

Fig 7.1.5:2 Instruktionsblandningar enligt Solomon och Arbuckle

Låt oss betrakta detta sätt att representera "effektivitet" för databehandlingssystem: Solomon avsåg belysa effektivitet för representativa "hela" datorsystem, och använde även i analysen systemkostnader i enlighet härmed, nämligen baserade på angivelser i Adams, ref (46). Det är minst sagt tveksamt om de fyra bearbetningarna ovan kan anses spegla ett datorsystems hela förmåga. Snarast framträder en dominans för ren processorkapabilitet.

Tyngdpunkten på belysning av internbearbetning i instruktionsblandningar är egentligen naturlig, eftersom stora skillnader föreligger mellan olika systems sätt att handha datatransporter till yttre minnen. Det är därför svårt att finna representativa "blandningar" för tillämpningar med t ex tunga massdatatransporter. Emellertid vet vi att

i moderna databehandlingssystem just samspelet mellan centralenheten och de yttre minnena har fundamental betydelse för ett stort antal tillämpningstyper. Av detta skäl blir användning av instruktionsblandningar för att jämföra system ingalunda fullständigt belysande. Bubenko kommenterar detta i (47), sid 49:

> "Att jämföra datamaskiners interna kapacitet genom instruktionsblandningar är enligt författarens åsikt endast en något mer "sofistikerad" metod än en enkel jämförelse med hjälp av t ex additionstider. I dessa instruktionsblandningar kan effekten av speciella, tidsbesparande instruktioner ej medtagas (t ex för bearbetning av brytsignaler, in/ut-matning, editering m m)".

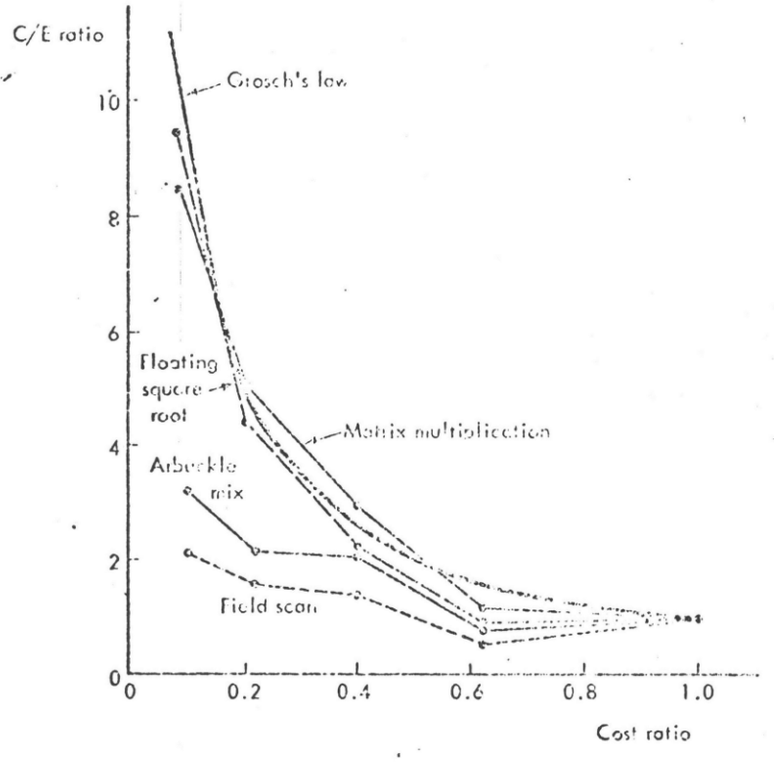
Inte ens internbearbetningskapacitet kan sålunda framgångsrikt avspeglas i instruktionsblandningar för de fall system skall jämföras som arbetar med väsentligt olika instruktionsmängder. Calingaert har i (48), sid 15, angett ett exempel härpå:

> "The members [of a group of experienced system engineers] were asked to specify the time in microseconds on System/360 Model 40 for the compare class of instructions, given only the fact that the original mix was based on the 7090 ..... The ten answers ranged from 11.88 to 30.66 with a mean of 21.5 and standard deviation of 7.0".

Vi vill här påpeka att mer detaljerad diskussion om möjlig relevans och användbarhet för instruktionsblandningar följer senare i detta arbete, i samband med belysning av hjälpmedel för egenskapsutvärdering.

Trots ovan nämnda brister kan de resultat Solomon nådde anses av visst intresse. (En metod för belysning av ett problem är nästan alltid av ett visst intresse tills någon väsentlig mer framgångsrik metod lanseras.) Följande sammanfattning av Solomons resultat ges av Sharpe i (28),

sid 318 där både abscissa och ordinata är skalade till 360/75 = 1.000:



Model	Cost ratio	C/E Matrix Mult.	C/E Fltg. Sq. Root	C/E Arbuckle Mix	C/E Field Scan
75	1.000	1.000	1.000	1.000	1.000
65	.625	1.127	0.900	0.975	0.687
50	.400	2.879	2.162	2.250	1.426
40	.213	4.880	4.628	2.248	1.053
30	.100	8.595	9.597	3.238	2.143

Source: Based on data in Solomon, "Economies of Scale and the IBM System/360," *Communications of the ACM*, June 1966.

Fig 7.1.5:3 Mätningar av effektivitet enl Solomon

En motsvarande studie, som liksom Solomons bygger på förmodat relevanta instruktionsblandningar, i detta fall en "tekniskt/vetenskaplig" och en "administrativ", har utförts av Knight, ref (49). Denna är intressant i sammanhanget bl a därför att samma systemmodeller betraktades. Knight's konfigurationer skiljer sig emellertid något från Solomons, dock knappast på ett avgörande sätt, jfr fig 7.1.5:1. För jämförelse ger vi även Knight's resultat (ur Sharpe (28), sid 320):

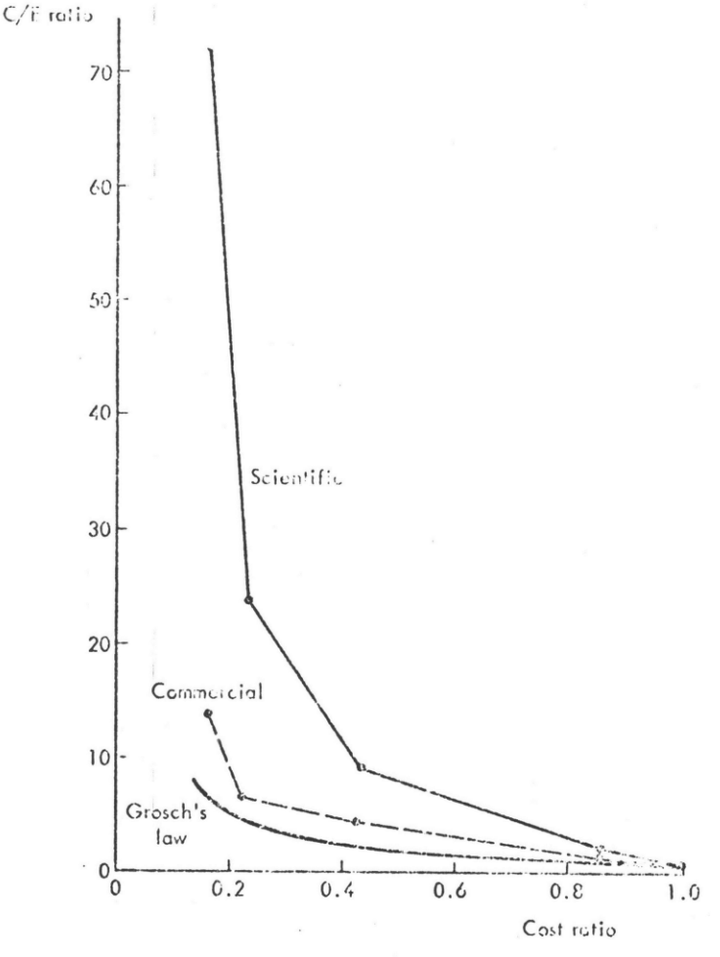


Fig 7.1.5:4 Mätningar av effektivitet enl Knight

Vi kan konstatera att Solomon's och Knight's resultat skiljer sig något från varandra. Emellertid föreligger en intressant överensstämmelse: väsentligt större storleksekonomi (economy of scale) tycks föreligga för tekniskt/vetenskapliga bearbetningar än för administrativa. Detta resultat överensstämmer dessutom med resultatet ur en studie av Allerdice, Carl och Chartrand, utförd under ledning av Sharpe, och kommenterad i (28), sid 321:

> "Far greater economies of scale are present for matrix inversion, which relies greatly on central processor capability. Finally, economies of scale are obtained for data-processing jobs (10 % file update and sorting), but they are of smaller magnitude".

Ur de relaterade undersökningarna, samt andra av liknande typ, kan vi draga följande slutsats: viss storleksekonomi (economy of scale) kan antas existera, dock att denna endast kunnat visas giltig för vissa schabloniserat avbildade tillämpningar. Resultaten tyder uppenbart på att effektivitetsrelationerna är i hög grad beroende av tillämpningsmiljön. Detta styrker omedelbart vår hypotes i detta arbetes avsnitt 1.

Vi avslutar diskussionen av Grosch's lag med att fästa uppmärksamhet på en intressant tankegång, som rör lagens eventuella giltighet. Trots beroende av tillämpningsmiljön föreligger sannolikt en viss relevans i lagen under förutsättning att tillämpningsmiljön preciseras avsevärt hårdare än i Grosch's utsaga. Under dessa förutsättningar, och med hänsyn till att så stora förändringar både på maskin- och kostnadssidorna inträffat sedan lagen formulerades (slutet av 1940-talet), har Behrens i (50) frågat sig om väsentligt yttre faktorer medverkar till lagens (relativa) giltighet:

> "Certainly we can assume that if Grosch's law was true at the time of its first utterance the present prices have not been established to simply perpetuate its

truthfulness."

.....

"The constancy of this law suggests that there are more basic factors involved, and that the efforts expended on examining Grosch's law may very well be simply verification of factors which extend beyond the computer industry."

.....

"If in an industry technological and economic factors similar to those working in the computer industry exist, and quantitative performance characteristics vs. costs are determined, then Grosch's law will apply."

Även om vi funnit anledning konstatera att Grosch's lag ej är generellt giltig, och framför allt är beroende av tillämpningsmiljön på ett sätt som Behrens ej betonar ovan, synes mycket tala för att för den typ av giltighet lagen tycks besitta, kan mycket väl Behrens' antagande vara giltigt. Det väntar emellertid på empiriskt belägg (eller motbevis) från miljöer utanför databehandlingsvärlden.

7.1.6 Effektivitet i program- och personvaror

I anslutning till våra resonemang om storlek och effektivitet kan vi fråga oss vilken plats program- resp personvaror intar i dessa avseenden. Gäller där liknande förhållanden, att under vissa förhållanden, i en väldefinierad miljö, effektivitetsvinsten i vissa fall kan uppnås?

Det första klarläggandet som erfordras för att kunna besvara denna fråga gäller möjligheterna att explicit kunna särskilja program- resp personvaror från maskinvaror. Vi har kunnat lägga märke till att programvaror vid slutet av 1960-talet av flera leverantörer börjat kopplas loss från maskinvarorna i offert- och upphandlingssammanhang. Denna utveckling utgår uppenbarligen från det evèntuella faktum att dessa delar går att betrakta (och prissätta) för sig.

Det kan anses tveksamt om en klar skiljelinje kan dras mellan, exempelvis, program- och maskinvaror. I begreppet programvaror brukar vi inkludera operativsystem och kompilatorer. Det är uppenbart att vissa delar av ett operativsystem (bl a inläsningsrutiner för övriga styrfunktioner) måste tillhandahållas till "en maskinvara" för att denna skall kunna bringas att arbeta. För de fall dessa rutiner tillhandahållits i programvaruform, vilket hittills nära nog alltid varit fallet, ser vi skäl att i maskinvaran inkludera dem. Detta innebär att skiljelinjen blir diffus. Ett annat skäl för denna utsuddning av den eventuella gränsen mellan program- och maskinvaror utgörs av det faktum att utvecklingen rörande s k mikroprogrammering leder fram mot ett läge där väsentliga delar av bl a operativsystemet(-en) tillhandahålles i maskinvaruform. Emedan den utsträckning, i vilken mikropgroam tillhandahålls, är direkt leverantörsberoende, finner vi det synnerligen svårt att kategoriskt klassa vissa instruktioner som tillhörande programvaror, och andra som maskinvaror.

På personvarusidan är läget av motsvarande art. Vi kan inte entydigt skilja dem från program- och maskinvaror utan att specificera både leverantör och tillämpningsmiljö. Vi finner sålunda skäl att säga att redan definition av begreppen programvaror och personvaror är miljöberoende.

Vi startade vår diskussion av Grosch's lag med förmodandet att lagen torde behandla förhållandevis "hela" system. Efter hand nådde vi slutsatsen att inte endast behövde vi begränsa diskussionen till den närmaste omgivningen till centralenheten i ett maskinsystem, vi behövde dessutom precisera aktuell användarmiljö". Vad gäller program- och personvaror torde situationen i likhet härmed vara den att inte bara föreligger problem med avgränsning av begreppen från varandra, vi finner dessutom skäl att anta ett miljöberoende.

Redan av dessa skäl förefaller det svårt att föra en meningsfull diskussion om "storlek och effektivitet" avseende program- resp personvaror. Dessutom måste vi emellertid notera utomordentliga svårigheter vid klargörandet av innebörden i begreppen "storlek" och "effektivitet". Skulle vi i "storlek" inrymma "generalitet", "flexibilitet" eller "kompatibilitet" eller något annat? Och vad avser vi med "effektiva" programvaror, t ex? Program med hög exekveringshastighet eller snabbkompilerande översättningsprogram? "Effektiva" på vilket datorsystem?

Det är uppenbart att svårigheterna är överväldigande vid försök att expandera Grosch's lag (eller någon liknande) till programvaru- eller personvarusidan.

W.J.Erikson har i (35), sid 109 ff, behandlat problemet med olika operativsystems inverkan på viss effektivitet i användning av förhållandevis "kompletta" datorsystem. Även om detta komplicerade problem inte på detaljnivå berör vårt i detta arbete aktuella ämne, vill vi helt kort relatera några av de synpunkter och resultat som där beskrivs.

Erikson betraktar datorsystemets belastning, d v s aktuella applikationsprogram och deras behandling i systemet, som konstanter, och studerar vilka totala kostnader fyra olika operativsystem (inkluderande skilda umgängessätt med systemet) leder till. Sålunda konstateras:

> "This study utilized analytical models that considered total cost (user cost plus computer system cost) as the variable while keeping throughput and turnaround time as constants."

Den situation som behandlas kan beskrivas med följande arbetsutsaga:

> "Configurate one computer as required for each of the four operating systems and compare their costs when each has an optimal user load."

Det besvärliga i att finna den optimala belastningen, samt även antydan om vad som avses därmed, kommenteras med att systemen betraktas vid "a throughput level that results in its total cost per job being a minimum". Härefter övergås till att variera belastningen, och undersöka vid vilka omständigheter den lägsta totala kostnaden uppnås. Utan att gå in på detaljerna i Eriksons arbete kan vi konstatera att resultaten på ett uppenbart sätt speglar det faktum att varken operativsystem för time-sharing, centraliserad satsvis bearbetning, avlägsen (remote) satsvis bearbetning eller för ett personligt dedikerat system uppvisar lägsta totalkostnad för samma bearbetningar. "Bästa" system beror direkt på bearbetningarnas art, huruvida jobben är av interaktiv art eller ej, hur hög throughput som betraktas m m. Vi finner alltså ett uppenbart miljöberoende. Två av Eriksons slutsatser ur arbetet innebär:

> "User costs are an important element in the total cost of programs developing or the operation of them and should be considered in the choice of a system."
>
> ....
>
> "It is not valid to argue that one type of operating system is better than any other for running completed programs or developing new programs. One must determine

> the particular range of conditions under which one system is better than another."

Eriksons användning av begreppet "optimal user load" finner en motsvarighet i en utsaga av Sharpe i (28), sid 281, där han à propos val av databehandlingssystem talar för

> "an approach, in which each of several alternative configurations is rated on the basis of its overall value and cost if used in the best manner (i.e. "best for the configuration in question)."

Det är utan tvekan önskvärt att vid jämförande analys betrakta system under någon sorts "optimal"belastning, p g a att därmed ett antal osäkerhetsfaktorer bringas ur bilden. Det måste emellertid betonas vilka stora svårigheter som är förknippade med att finna denna belastning, eller detta bearbetningssätt, speciellt som i praktiska situationer ett "exekutivt optimum" (där bl a tillgänglig tid spelar in) är av större intresse än det sanna optimet.

Vi avslutar detta avsnitt med att relatera en slutsats ur en annan studie av programvarusystem plus personvaror, utförd av J.C. Hammerton, och beskriven i (51), sid 70 ff. Det behandlas där situationer som uppkommer vid införande av time-sharing-service i ett antal företagsmiljöer. Vad som främst bedöms intressant är total kostnad, bl a som funktion av antal av systemet direkt berörda befattningshavare inom företaget. Beträffande totala kostnader nås resultat som liknar här tidigare relaterade, men dessutom kan Hammerton konstatera ett annat miljöberoende:

> "Where an organization has a large clerical staff doing essentially manual - i. e. non-machine aided

> information processing - the economic justification for the time-shared system is well founded, particularly if the organization is prepared to go all the way with the time-shared service. Where the organization has kept its clercial staff to a minimum, the economic justification is nonexistent or unimpressive."

Här finner vi sålunda en miljöpåverkan som icke är enbart kostnadsberoende.

7.2 Egenskaper på lägre nivåer

De resonemang vi i detta kapitel hittills fört har rört de eventuella möjligheterna att kunna betrakta egenskaper för databehandlingssystem gruppvis, dvs att direkt - utan ytterligare nedbrytning - kunna bedöma bidrag till måluppfyllelse för grupper av egenskaper. Vi har funnit det svårt att utföra dylik gruppering på ett måloberoende sätt, enär beroendet av användarmiljön visat sig av stor betydelse för grupperingarnas art.

Vi tvingas därför betrakta egenskaper på lägre nivåer. Detta är i sig mindre lyckligt, av det skälet att därmed möjligheterna att målrelatera systemen minskar, eller åtminstone görs mindre säkra i samband med vanligen förekommande mål. Ju större avstånd mellan två punkter i ett universum, desto mindre noggrannhet erhålls vid mätning av avståndet (eller relationen) dem emellan, under förutsättning av fixerade mätdon. Vi konstaterar därför ett dilemma: för att erhålla stor noggrannhet avseende egenskapernas karaktäristika behöver vi betrakta dem på låg nivå (höggradigt nedbrutna), men för att kunna relatera dem avseende bidrag till måluppfyllelse vill vi - för att nå god noggrannhet i bedömningarna - ej bryta ner dem för långt. Lösningen tar ofta formen av en kompromiss, och de egenskaper som vi nedan diskuterar, befinner sig därför ingalunda på "lägsta tänkbara" nivå (f ö ett relativt begrepp), men dock i huvudsak relativt nära själva datorerna i databehandlingssystemen.

7.2.1 Egenskapernas roll i offertarbetet

Det kan vara på sin plats att kommentera en utveckling avseende databehandlingstjänster som gör sig påmind vid 1970-talets början. Hittills (t o m 1960-talet) har i stort sett varje upphandling av databehandlingsutrustning

löpt efter mallen att den potentielle köparen utsänder sin offertförfrågan till ett antal leverantörer, vari preciseras på så långt möjligt "elementär" nivå vilka egenskaper det önskade systemet skall besitta. Den nivå som därvid ofta varit aktuell har inneburit karaktäristika på "cykeltidsnivå" (motsv). Efter det att offerter inkommit har utvärderingsarbetet bedrivits av (den potentielle)köparen själv. Detta arbete har ofta visat sig betungande för honom, och han har inte alltid varit, eller ansett sig vara, i besittning av erforderlig expertkunskap därför.

En utveckling som vi nu börjar skymta är att mer och mer av analysarbetet upp till tillämpningsnivå övertas av konsulter eller leverantörer av speciella slag. Dessa kan bättre överblicka den alltmer ökande floran av möjliga leverantörer i varje enskilt fall, speciellt vad gäller tillhandahållande av delar av system. (En starkt ökande konkurrens vad gäller tillhandahållande av enskilda systemelement, såväl på maskin- som på programvarunivå är att vänta).

Denna utveckling innebär att den potentielle köparen frikopplas från väsentliga delar av utvärderingsarbetet, i och med att han beställer hela system som skall utföra väldefinierade tjänster på tillämpningsnivå, snarare än speciella datorsystem som möjligen - i bästa fall - kan lösa hans problem. Härmed sker en överflyttning av en väsentlig del av ansvaret för att systemet verkligen kan lösa de aktuella tillämpningsuppgifterna, från köparen till konsulten/leverantören. Köparen beställer därmed, och får betala för, ett system som med uppgivet stor sannolikhet/säkerhet löser hans problem. Ansvaret för framgång resp risken för misslyckande överflyttas då till konsulten/leverantören, som i medvetande om detta säkert kommer att veta att ta betalt därför.

För köparen innebär denna utveckling (i specialiserande riktning) det positiva att han slipper merparten av de besvärliga tekniska problemen, bl a i utvärderingsfasen, och får en viss garanti för att systemet verkligen kommer att fungera på avsett sätt. Det negativa för honom ligger i att denna "garanterade framgång" kommer att kosta mera, samt - och inte minst - att ökade krav ställs på hans förmåga att kunna beskriva vad systemet skall kunna utföra i tillämpningsmiljön. Denna beskrivning måste han ge i precisa termer, något som måste anses svårt, speciellt vad avser tillämpningar som vid upphandlingstillfället ännu inte är preciserade, eller kan preciseras i tillräcklig detalj.

Det bör betonas att köparen icke bör lägga ut den principiella beskrivningen av det önskade systemets arbetsuppgifter. Endast han har tillräcklig kännedom härom. I analogi med detta framhålls i (52), sid 3:

> "The user should be responsible for doing the system analysis and creating the design of the new system. The system engineer should not sit in on the new design. Instead, he should work from design specifications developed by the user".

Väsentliga delar av utvärderingsarbetet kommer sålunda troligen småningom att övertas av specialister, som tillhandahåller förhållandevis kompletta tillämpningssystem, inklusive driften därav. Köparen av dylika system bör dock sannolikt i längden vinna på att hålla sig informerad om de väsentliga huvuddragen i det arbete som därmed utläggs utanför företaget.

7.2.2 Beskrivning av typiska egenskaper

I detta avsnitt vill vi ange en uppsättning konkreta kännetecken för databehandlingssystem. Dessa kommer vi att beskriva med hjälp av en verbal uppräkning av kommenterande frågor, användbara t ex när kännedom om karaktäristika för ett system skall inhämtas från en (eller flera) leverantör(er) i offertsammanhang.

Vi har tidigare funnit skäl betona svårigheterna vid försök att finna generella egenskaper för databehandlingssystem, oberoende av tillämpningsmiljö. I enlighet härmed är självfallet nedanstående lista icke lämpad att vara allmänt användbar. Lämpligt urval ur listan får i varje enskilt fall göras, och - framför allt - substansiella kompletteringar behöver göras som beskriver aktuell funktionsmiljö så detaljerat som möjligt.

Den kommenterande listan måste således, såväl av principiella som av praktiska skäl, anses som ofullständig.

Operativsystem

1. Beskriv de bearbetsformer som är möjliga med systemet, samt huvuddragen av dess inre arbetsprinciper.
2. Beskriv styrspråket.
3. Finns villkorlig länkning av program? Beskriv övriga användarprogrammerade segmenteringsprinciper.
4. Vilka koder kan accepteras på databärande media?
5. Vilka minnesbehov har den i denna tillämpningsmiljö aktuella versionen av operativsystemet?
6. Hur stort minnesutrymme för in/ut-köer rekommenderas?
7. Kan operativsystemets administrationstid (overhead) anges för några givna belastningar?
8. För vilka språk (inkl versioner därav) finns kompilatorer/interpretatorer i drift? Sedan när?
9. Beskriv det standardiserade debiteringssystemet. Hur flexibelt (förändringsbart) är detta system?
10. Vilka faciliteter för reetablering och backup efter breakdown på olika nivåer existerar? Beskriv aktuella checkpoinkt/restart-funktioner.
11. Beskriv aktuella roll out/roll in-funktioner.
12. Beskriv aktuell form av multiprogrammering vid olika användningssätt av systemet.
13. Vilka faciliteter för undvikande av låsning erbjuds?
14. Beskriv systemets möjligheter avseende program/data-oberoende.
15. Vilka sekretessmöjligheter erbjuds?
16. Vilket internt tidtagningssystem används inom maskinsystemet, samt vilka åtkomstmöjligheter därtill existerar från olika användarprogramnivåer?
17. Vilka principer för primär- resp sekundärminnesutnyttjande används?
18. Kan kommunikation från flera låghastighetsterminaler pågå samtidigt till samma program?

Operativsystem forts

19. Använder systemet virtuell minnesteknik? I så fall, hur?
20. Arbetar systemet med dynamisk relokering av användar- och systemprogram?
21. Kan ett program öka resp minska sitt minnesbehov dynamiskt under exekvering?
22. Tillåter systemet simultanbehandling av flera on-line-transaktioner (multithread)? Samtidighet med annan bearbetning?
23. Hur genereras operativsystemet?

Kompilator (för språket X)

1. Ge exempel på kompilatorns objektkodseffektivitet enligt angivna normer.
2. Ge exempel på kompileringshastigheter i några givna fall (nollprogram, testprogram).
3. Finns olika kompilatorer för språket X? Vad skiljer dem åt?
4. Beskriv kompatibilitet med existerande standards för språket X.
5. Hur minneskrävande blir programmet Y (se ref) efter kompilering?
6. Hur minneskrävande är aktuell kompilator på primär- resp sekundärminne?
7. Vilka möjligheter finns att sammanlänka program skrivna i andra språk med program i språket X?
8. Vilka debug- resp tracingfaciliteter existerar?
9. Vilka möjligheter finns att hantera oberoende procedurer, skrivna i språket X?
10. Producerar kompilatorn relokerbar kod?

Applikationsprogram

1. Beskriv existerande editeringssystem. Kan såväl källkod som objektkod och data editeras? Hur?
2. Beskriv existerande filhanteringssystem. Vilka access- och lagringsmetoder är möjliga? Empiriska erfarenheter?
3. Finns översättningsprogram för program i språket Y från källkod för leverantör Z till er källkod?
4. Vilka mediakonverteringsprogram erbjuds?
5. Vilka sorteringsprogram erbjuds?
6. Vilken typ av rapportgenerator erbjuds?
7. Finns program för uppsamling av intern belastningsstatistik för systemet?
8. Vilka övriga serviceprogram tillhandahålls?

Synpunkter av personvaruart

1. Vilka driftserfarenehter från andra installationer kan redovisas?
2. Erfarenheter från inlärning och löpande hantering av styrspråket?
3. Erfarenheter från filhantering på aktuellt system?
4. Erfarenheter från programspråksskillnader?
5. Erfarenheter från användning av maskinvarukoder?
6. Erfarenheter från programvarudiagnostik såväl vid kompilering som vid exekvering?
7. Erfarenheter från läsbarhet av ordinarie kompileringsutskrifter?
8. Hur många raders utskrift på radskrivare erhålls vid normal kompilering och exekvering av ett nollprogram?
9. Erfarenheter vad gäller operativsystemets hjälpmedel för system- och programproduktion?

Processor(er)

1. Är multipla processorer maskinvarumässigt möjliga?
2. Finns maskinvaruinstruktioner för hantering av delar av ord/tecken (bitmanipulation) ?
3. Ordlängd? Fast eller variabel? Paritetsbitar?
4. Antal indexregister? Funktion och användbarhet från aktuella (även högre) språk?
5. Teckenstorlek?
6. Accessvidd (antal bitar/access)?
7. Cykeltid?
8. Beskriv det avbrottssystem som tillämpas.
9. Beskriv den maskinvarumässiga instruktionsmängden.
10. Ange tidsåtgång för nämnda instruktioner, såväl min som max och genomsnitt.

Primärminne(n)

1. Storlek i ord (motsv), maximalt resp minimalt?
2. Erbjuden minnesstorlek?
3. Cykeltid?
4. Hierarkisk minnesstruktur? Olika cykeltider?
5. Finns möjlighet att ansluta yttre kärnminnen (motsv)?
6. Ange aktuella principer för minnesskydd.

Sekundärminnen

1. Storlek i ord (motsv)?
2. Överföringshastighet, max, min och genomsnitt?
3. Accesstid, max, min och genomsnitt?
4. Blockstorlek?
5. Läsning/skrivning i flera riktningar?
6. Bandbredd (för magnetband)?
7. Möjliga bandlängder (för magnetband)?
8. Packningstäthet?
9. Start/stopp-tid (för magnetband)?
10. Kompatibilitet med andra magnetbandsbredder?

Kanaler

1. Beskriv kanalsystemets arbetssätt, speciellt med tonvikt på möjlig simultanitet.
2. Existerar multiplexorenheter? Max antal?
3. Existerar selektorenheter? Max antal?
4. Beskriv aktuell buffertteknik.
5. Ange de ev olika kanalenheternas överföringshastigheter.
6. Ange max minnesutrymme i resp enheter.
7. Vilka yttre enheter kan anslutas till resp kanaler?

Övriga yttre enheter

1. Beskriv prestanda för erbjudna yttre enheter.
2. Vilka dataöverföringskontroller tillämpas?
3. Beskriv systemet för kommunikation med operatörerna.
4. Finns emulatorer för körning av objektprogram från maskinsystem Y? Vilken effektivitet gäller för dessa?

Kommunikationsutrustning

1. Beskriv föreslagen utrustnings arbetssätt.
2. Vilket max antal linjer kan anslutas?
3. Vilket avkänningssystem tillämpas för linjerna, via avbrott eller via polling?
4. Vilken kapacitet har linjerna?
5. Vilka typer av terminaler kan anslutas?

Pris

1. Ge köpepriset samt hyrespriset för aktuell konfiguration.
2. Vilka rabatter kan diskuteras?
3. Kan viss maskintid på aktuell maskin återköpas av leverantör-en/erna?
4. Vilket restvärde uppskattar Ni att den aktuella konfigurationen har efter full altuell avskrivning?

Leveranstid

1. Vilken är leveranstiden för aktuell utrustning?
2. Ange leveranstider för eventuella möjliga systemutbyggnader.
3. Vilken samtidighet gäller för leverans av program- och maskinvaror?
4. Vilka ersättningar för försenad leverans är Ni villig att erbjuda?

Underhåll och service

1. Beskriv arten och omfattningen av det underhåll som krävs för systemets drift samt den löpande service som lämnas i samband därmed. Detta gäller såväl program- som maskinvara.
2. Vilken tillgång erbjuds till högkvalificerad systempersonal?

Dokumentation

1. Ange vilken dokumentation som beskriver aktuellt system.
2. Vilken därefter löpande dokumentation om systemändringar tillhandahålls?
3. Vilka erfarenheter från andra användare av aktuellt system kan Ni hänvisa till vad gäller dokumentationens lämplighet i utformning och tillämpbarhet?

Tillförlitlighet

1. Vilken backup kan erbjudas?
2. Ange tillgänglighetsmått för dels hela systemet (inkl programvara) och dels för varje aktuell enhet därav. Måttet definieras som

$$T = MTBF / (MTBF + MTTR)$$

där MTBF = mean time between failures
MTTR = mean time to repair

Utbyggnadsmöjligheter

1. Vilka programvaru- och maskinvarumässiga utbyggnadsmöjligheter existerar?
2. Vilka fundamentala restriktioner gäller?

Omläggning

1. Vilken omläggningsservice erbjuds, utgående från angivna önskemål?

Utbildning

1. Ange erbjuden för systemets hantering relevant utbildningsservice.

Installation

1. Beskriv erforderliga installationsegenskaper, berörande:
   - fysiskt utrymme
   - effektbehov/frekvens
   - temperatur
   - luftfuktighet
   - luftfiltrering
   - vibrationstoleranser
   - golvtryck

Yttre faktorer

1. Vilket rykte kan maskinsystemet anses ha på marknaden? Vilka belägg härför finns?
2. Finns empiriska erfarenheter från systemets användning bekanta? Kan dessa redovisas?
3. Hur länge har det aktuella systemets samtliga delar funnits i löpande drift?
4. Hur många installationer av det aktuella systemet existerar i Europa/USA?
5. Existerar organisation av användare av detta system?

Det betonas att detta arbete härmed ej är avslutat. I fortsättningen kommer ingående att diskuteras bl a vilka hjälpmedel som existerar för att ur kännedom om ett datorsystems egenskaper på låg nedbrytningsnivå kunna draga slutsatser om dess uppträdande i en miljö som definieras i termer på högre nivåer. Härvid kommer bl a att betraktas:

- analytiska belastningsberäkningar
- användningsbarhet av instruktionsblandningar
- simulering
- fysiska belastningsmätdon
- olika former av testkörningar

m m

Litteraturreferenser
(i ordning efter första uppträdande referens i texten)

1. Brandon, D — Föredrag refererat i EDP Weekly, nov 66
2. Grosch, H — Referat i Europarådet, Committee on science and technology, Subcommittee on data processing, AS/Science/Computer (22),25, jan 71
3. Gyllenhammar, P — Mot sekelskiftet på måfå R Larsson AB 70
4. Alt, F L — Computers - Past and Future
5. Laska, R M — Keeping your computer out of the till Computer Decisions, maj 70
6. Langefors, B — Theoretical Analysis of Information Systems Studentlitteratur 69
7. Bubenko, J — CADIS WP 28, juli 70
8. Langefors, B — Systemering II, utdrag ur IB-ADB-rapporter 67/2, 68/10, 68/14, 68/15, 68/32, 69/1, 69/2, 69/3, 69/6, 69/7, 69/8, 70/12 Inst för informationsbehandling, KTH
9. Langefors, B — System för företagsstyrning Studentlitteratur 68
10. Johnsen, E — Studies in multiobjective decision models Studentlitteratur 68
11. Langefors, B — Goal structure, IB-ADB 68/23 Inst för informationsbehandling, KTH
12. Karplus, W J — On-line computing
13. Bubenko - Ohlin — Introduktion till operativsystem Studentlitteratur 71
14. Martin-Norman — The computerized society Prentice-Hall 70

15. Kanter, J — Management guide to computer system selection and use
Prentice-Hall 70

16. Langefors, B — Decision and control in a multi-object environment. IB-ADB 70/23
Inst för informationsbehandling, KTH

17. Hughes, C L — Goal setting
American Management Association 65

18. Langefors, B — Management Information Systems Design
IAG Quarterly 1969, Vol 2, nr 4

19. AFIPS Conf Proc 1964, vol 25 — Joslin/Mullin: Cost-value technique for evaluation of computer systems proposals

20. Joslin, E O — Computer Selection
Addison-Wesley 68

21. Green, B — Attitude measurement
Handbook of Social Psychology
Addison-Wesley 54

22. Torgerson, W S — Theory and methods of scaling
J Wiley 58

23. Langefors, B — Systemevaluering. IB-ADB 69/9
Inst för informationsbehandling, KTH

24. McKean — Efficiency in government through systems analysis. J Wiley & Sons

25. Swedner, H — Sociologisk metod
Gleerups 66

26. Hidas, P — Computer evaluation, methods and experiences. Universitetsförlaget 70

27. Boalt-Jonsson — Att undersöka beteende
Natur och Kultur 66

28. Sharpe, W F — The economics of computers
Columbia University Press 1969

29. AFIPS Conf Proc. 1964, vol 25 — Rosenthal: Analytical technique for automatic data processing equipment acquisition.

30. Scharf, T — Weighted ranking by levels (motsv) Proc of Norddata 68

31. Scharf, T — Weighted ranking by levels IFIP IAG Quarterly 69

32. Sussman, B — Peut-on choisir en tenant compte de critères multiple? Une méthode (Electre) et trois applications. (Tills med Buffet, Grémy och Marc) Metra, France 1967

33. Bright, J R (red) — Technological forecasting for industry and government, Special report, International magazine

34. Mathews, M V — Choosing a scientific computer for service, Tidskr. Science, 5 juli 1968

35. Weinwurm, G F — On the management of computer programming, Auerbach 1970

36. Orlicky, J — The successful computer system, McGraw-Hill 1969

37. Schneider, C — Kleincomputer oder Rechenzentrum? Betriebwirtschaftlicher Verlag, Wiesbaden 1968

38. Zack, R A — Minicomputer applications - do they imply something for everybody? Computers and Automation, dec 1969

39. Berkely, E C — Minicomputers, Computers and Automation, dec 1969

40. Clapp, L C — A time-sharing computer terminal vs the small computer, Computers and Automation, dec 1969

41. de Greene, K B — Systems psychology, McGraw-Hill 1970

42. Amdahl, L D — Architectural questions of the seventies, Datamation, jan 1970

43. Langefors, B — Computer networks, IB-ADB 68/31 Inst för Informationsbehandling, KTH

44. Solomon, M B — Economies of scale and the System/360, CACM, juni 1966

45. Arbuckle, R A — Computer Analysis and thruput evaluation, Computers and Automation, jan 1966

46. Adams, C — Computer characteristics Quarterly, mars 1965

47. Bubenko, J — Databehandlingsteknik, Studentlitteratur 1967

48. Calingaert, P — System performance evaluation: survey and appraisal, CACM, jan 1967

49. Knight, K — Evolving computer performance, 1962-67, Datamation, jan 1968

50. Behrens, E A — Why does Grosch´s law continue to be true? CACM, nov 1966

51. Hammerton, J C — Business time-sharing: user economics, Datamation, juni 1969

52. Canning Publ Inc — Selecting new products and services, EDP Analyzer, juli 1970

Förteckning över figurer

Fig	Innehåll	Sid
2:1	Människa-maskin	6
3:1	Schematisk bild av systemutvärdering	14
4.1:1	Mål på olika nivåer	19
4.1:2	Måls miljöberoende	22
4.2:1	Måldynamik	25
4.2:2	Planering avseende datorkapacitet	26
4.6:1	Operationell del av målstruktur	39
4.6:2	Exempel på målbehovsanalys	41
5.1:1	Ambitionsnivå	47
5.2:1	Ambitionsnivåinnehåll	51
6.2.2.1:1	Systemrelatering till mål	67
6.2.2.2:1	Oviktad poängsättning	70
6.2.2.3:1	Viktad poängsättning (trenivåigt ex)	73
6.3.1:1	Dokumentation av vunna parvisa jämförelser	80
6.3.2:1	Viktade parvisa jämförelser	82
7.1.5:1	Typiska relativa kostnader för IBM-system	115
7.1.5:2	Instruktionsblandningar enligt Solomon och Arbuckle	116
7.1.5:3	Mätningar av effektivitet enl Solomon	118
7.1.5:4	Mätningar av effektivitet enl Knight	119

Litteraturreferenser

(i bokstavsordning)

Adams, C	Computer characteristics Quarterly, mars 1965
AFIPS Conf Proc 1964, vol 25	Joslin/Mullin: Cost-value technique for evaluation of computer systems proposals
AFIPS Conf Proc 1964, vol 25	Rosenthal: Analytical technique for automatic data processing equipment acquisition
Alt, F L	Computers - past and future
Amdahl, L D	Architectural questions of the seventies, Datamation, jan 1970
Arbuckle, R A	Computer Analysis and thruput evaluation, Computers and Automation, jan 1966
Behrens, E A	Why does Grosch´s law continue to be true? CACM, nov 1966
Berkely, E C	Minicomputers, Computers and Automation, dec 1969
Boalt - Jonsson	Att undersöka beteende Natur och Kultur 1966
Brandon, D	Föredrag refererat i EDP Weekly, nov 1966
Bright, J R (red)	Technological forecasting for industry and government, Special report, International Magazine

Bubenko, J	CADIS WP 28, juli 1970
Bubenko, J	Databehandlingsteknik, Studentlitteratur 1967
Bubenko/Ohlin	Introduktion till operativsystem, Studentlitteratur 1971
Calingaert, P	System performance evaluation: survey and appraisal, CACM, jan 1967
Canning Publ Inc	Selecting new products and services, EDP Analyzer, juli 1970
Clapp, L C	A time-sharing computer terminal vs the small computer, Computers and Automation, dec 1969
de Greene, K B	Systems psychology, McGraw-Hill 1970
Green, B	Attitude measurement Handbook of Social Psychology Addison-Wesley 1954
Grosch, H	Referat i Europarådet, Committee on science and technology, Subcommittee on data processing, AS/Science/Computer (22), 25, jan 1971
Gyllenhammar, P	Mot sekelskiftet på måfå R Larsson AB 1970
Hammerton, J C	Business time-sharing: user economics, Datamation, juni 1969
Hidas, P	Computer evaluation, methods and experiences Universitetsförlaget 1970

Hughes, C L
Goal setting
American Management Association 1965

Johnsen, E
Studies in multiobjective decision models
Studentlitteratur 1968

Joslin, E O
Computer selection
Addison-Wesley 1968

Kanter, J
Management guide to computer system selection and use
Prentice-Hall 1970

Karplus, W J
On-line computing

Knight, K
Evolving computer performance, 1962-67,
Datamation, jan 1968

Langefors, B
Computer networks,
IB-ADB 68/31
Inst för Informationsbehandling, KTH

Langefors, B
Decision and control in a multi-object environment
IB-ADB 70/23
Inst för Informationsbehandling, KTH

Langefors, B
Goal structure
IB-ADB 68/23
Inst för Informationsbehandling, KTH

Langefors, B
Management Information Systems Design
IAG Quarterly 1969, Vol 2, nr 4

Langefors, B
Systemering II, utdrag ur IB-ADB-rapp. 67/2, 68/10, 68/14, 68/15, 68/32, 69/1, 69/2, 69/3, 69/6, 69/7, 69/8, 70/12
Inst för Informationsbehandling, KTH

Langefors, B	Systemevaluering IB-ADB 69/9 Inst för Informationsbehandling, KTH
Langefors, B	System för företagsstyrning Studentlitteratur 1968
Langefors, B	Theoretical Analysis of Information Systems Studentlitteratur 1969
Laska, R M	Keeping your computer out of the till Computer Decisions, maj 1970
Martin - Norman	The computerized society Prentice-Hall 1970
Mathews, M V	Choosing a scientific computer for service Tidskriften Science, 5 juli 1968
Mc Kean	Efficiency in government through systems analysis, J Wiley and Sons
Orlicky, J	The successful computer system McGraw-Hill 1969
Scharf, T	Weighted ranking by levels IFIP IAG Quarterly 1969
Scharf, T	Weighted ranking by levels (motsv) Proc of Norddata 1968
Schneider, C	Kleincomputer oder Rechenzentrum? Betriebwirtschaftlicher Verlag, Wiesbaden 1968
Sharpe, W F	The economics of computers Columbia University Press 1969

Solomon, M B — Economies of scale and the System/360, CACM, juni 1966

Sussman, B — Peut-on choisir en tenant compte de critères multiple? Une méthode (Electre) et trois applications. (Tills med Buffet, Grémy och Marc) Metra, France 1967

Swedner, H — Sociologisk metod Gleerups 1966

Torgerson, W S — Theory and methods of scaling J Wiley 1958

Weinwurm, G F — On the management of computer programming Auerbach 1970

Zack, R A — Minicomputer applications - do they imply something for everybody? Computers and Automation, dec 1969

Appendix 1

Nedan redovisas en metod som är användbar som bakgrund för diskussion av planering av utvärderingsarbete. Metoden bygger på ett antal idealiserade antaganden, som måhända ej enkelt låter sig praktiskt konkretiseras. Trots detta bör redovisade samband mellan ingående variabler och funktioner kunna tjäna som belysning av aktuella planeringsproblem.

Följande antaganden görs:

1 Företagets mål är här ren kostnadsminimering.

2 Kostnaden för löpande drift av det i företaget existerande datorsystemet är Ke(t) kr/dag.

3 Kostnaden för löpande drift av det nya datorsystemet, i det ideala fall att det perfekt uppfyller erforderliga bearbetningskrav, är Kn(t) kr/dag.

4 Driftskostnaden i det fall det nya datorsystemet visar sig helt otillräckligt för att lösa de erforderliga uppgifterna är Q(t) kr/dag, d v s lika med kostnaden för att på annat sätt (externa servicebyråer etc) utföra bearbetningarna.

5 Den förväntade proportionen av de erforderliga bearbetningarna som kan utföras med det nya datorsystemet är p(t).

6 Tidsskalan väljs på så sätt att utvärderingsarbetet startas vid tidpunkten t = 0.

7 Kostnaden för utförande av utvärderingsarbetet är U(t) kr/dag

8 Inom företaget använd kalkylränta är r %.

9 Företagets verksamhet mellan tidpunkterna 0 och T betraktas.

10 Kostnadsfunktionen för inköp (alternativt hyra) av den nya datorutrustningen är S(t) kr.

11 Restvärdet vid tidpunkten T av det nya datorsystemet är Rv kr.

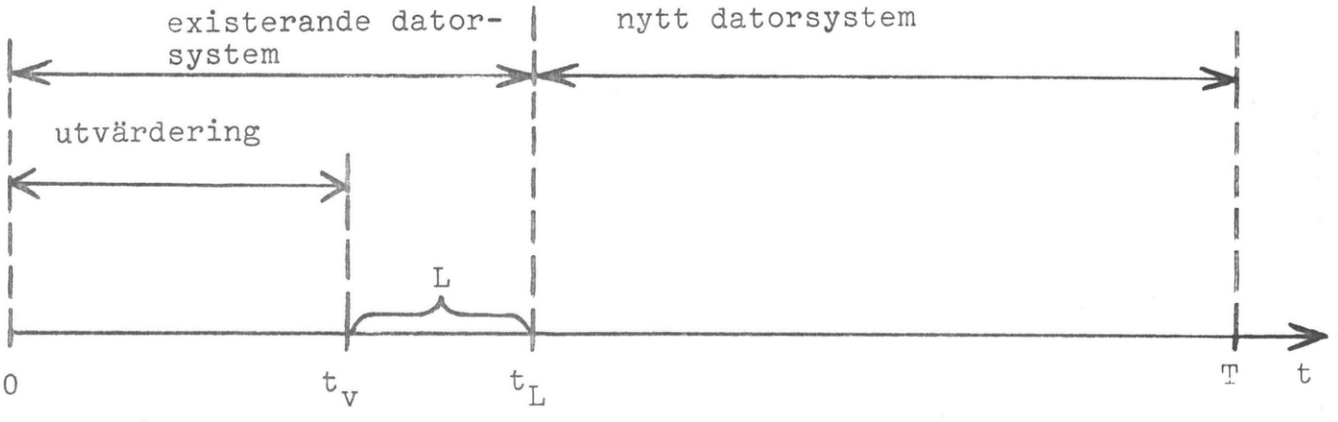
12 Tidpunkten då den planerade nya datorutrustningen levereras är t_L.

13 Tidpunkten då utvärderingsarbetet skall vara klart, och utrustningen skall beställas, är t_V.

Vi kan inledningsvis konstatera att då leveranstiden för utrustning "ofta" är nära nog lika lång (oberoende av leverantör), kan vi sätta denna till konstanten L. Vi har då

$$t_L = t_v + L$$

Den tidsmässiga utgångssituationen är:



Med dessa förutsättningar finner vi:

Nuvärdet av kostnaden för drift av den existerande utrustningen (mellan 0 och t_L):

$$\int_0^{t_L} Ke(t) \cdot (1 + r)^{-t} dt$$

ty driftskostnaden för tidsintervallet (t, t + dt) är Ke(t) · dt, och nuvärdet av denna kostnad är $Ke(t) \cdot (1 + r)^{-t} dt$.

Nuvärdet av kostnaden för bearbetningarna efter tiden t_L är

$$\int_{t_L}^{T} \left[p(t)\cdot Kn(t) + (1-p(t))\cdot Q(t)\right] \cdot (1 + r)^{-t} \cdot dt$$

Nuvärdet av anskaffningskostnaden för den nya utrustningen är

$$\int_{t_L}^{T} S(t) \cdot (1 + r)^{-t} \cdot dt$$

Nuvärdet av kostnaderna för utförande av utvärderingsarbetet:

$$\int_{0}^{t_v} U(t) \cdot (1 + r)^{-t} \cdot dt$$

Nuvärdet av restvärdet för det nya datorsystemet är

$$Rv \cdot (1 + r)^{-T}$$

Vi kan nu bilda den totala kostnaden

$$K = \int_{0}^{t_L} Ke(t) \cdot (1 + r)^{-t} \cdot dt + \int_{t_L}^{T} S(t) \cdot (1 + r)^{-t} \cdot dt +$$
$$+ \int_{t_L}^{T} \left[p(t)\cdot Kn(t) + (1-p(t))\cdot Q(t)\right] \cdot (1 + r)^{-t} \cdot dt +$$
$$+ \int_{0}^{t_v} U(t) \cdot (1 + r)^{-t} \cdot dt - Rv \cdot (1 + r)^{-T}$$

Under den givna förutsättningen att företaget eftersträvar <u>kostnadsminimering</u> söker vi det värde på t_v som ger den lägsta kostnaden K.

$$\frac{dK}{dt_v} = 0$$

Härur erhålls det optimala värdet på t_v.

Den erhållna ekvationen är icke alltid explicit analytiskt lösbar. I så fall kan lösningen erhållas numeriskt, t ex med Newton-Raphsons metod.

Låt oss kommentera de antaganden som gjorts. Numreringen nedan överensstämmer med den ovan givna numreringen för angivna förutsättningar.

1 Kostnadsminimering. Det enda mål som betraktas i samband med denna metod.

2 Kostnadsfunktion för drift av det existerande systemet. Denna funktion bör med tillfredsställande noggrannhet vara känd för företaget. Om funktionen ej kan ges sammanhängande integrerbara form för hela det aktuella tidsintervallet, kan den representeras av ett antal styckvis integrebara (t ex linjära) funktioner.

3 Kostnadsfunktion för drift av det nya datorsystemet. Denna funktion kan givetvis ej anses känd med tillfredsställande säkerhet. I stället antas här en lämplig, och rimlig, integrerbar funktion.

4 Kostnadsfunktion för bearbetning på annat sätt. Det kan vara rimligt att som Q(t) ansätta en funktion som inledningsvis (nära efter t_L) överstiger Kn(t), p g a igångsättningsproblem, men därefter sjunker till ett någotsånär konstant värde.

5 Den förväntade proportionen p(t). Denna funktion kan antas vara monotont ökande, och avspegla bl a förtroende för aktuell leverantör, erfarenheter från tidigare systembyten m m .

6 Tidsskalan. Är ej avgörande för analysen.

7 Kostnadsfunktion för utvärderingsarbetet. Denna funktion bör med tillräcklig säkerhet kunna uppskattas utgående från bl a kunskaper om arvoden för aktuell personal, rimliga kostnader för testkörningar m m.

8 Kalkylränta. Bör finnas tillgänglig inom företaget.

9 Tidsintervall. Det aktuella tidsintervallets längd väljs lämpligen så att tillfredsställande säkerhet för de aktuella antagna funktionerna inom internallet uppnås.

10 Kostnadsfunktion för anskaffning. Denna funktion, avspeglande köp eller hyra, får utseende som bestäms i förhandlingar med leverantören.

11 Restvärde. Detta bör med tillfredsställande säkerhet kunna ansättas, utgående från kunskaper om utrustningens värde för företaget, samt om utrustningars värde på den expanderande marknaden för begagnade datorutrustningar. För det fall restvärdet är konstant påverkas ej beräkningen av val av olika restvärden. Påverkas restvärdet av tiden, får en integral här ersätta den i kostnadsfunktionen aktuella termen.

12 Leveranstid. Kan oftast uppskattas med tillfredsställande säkerhet. Variationerna på marknaden kan ej anses stora.

13 Sökt tidsoptimum. Erhålls ur den angivna ekvationen.

Det bör noteras att den angivna analysen innehåller ett flertal funktionsuppskattningar, vilkas tillförlitlighet är osäker. Emellertid kan metoden ge en intressant belysning av hur känsliga parameter- resp funktionsansättningarna är, d v s vilken inbördes påverkan som erhålls vid olika antaganden.