

Teknikerens Integrerede Arbejdsplads
(TIA)

Teknikerens arbejde og Teknikerens Integrerede Arbejdsplads

Kjeld Schmidt

Dansk Datamatik Center
December 1987

Projektet er blevet finansieret af Teknologirådet

Indhold

Indhold.....	2
Forord.....	3
1. Indledning.....	4
2. Teknikerens arbejde.....	6
2.1. Det tekniske arbejdsdomæne.....	7
2.2. Det tekniske arbejdes problemrum	10
2.3. Grundfunktionerne i teknikerens arbejde.....	12
2.3.1. Konceptualisering: Søgning efter mulige alternative løsninger.....	13
2.3.2. Verifikation af evt. egnede alternative løsninger.....	13
2.3.3. Nedbrydning og koordination	14
2.4. Det tekniske arbejdes iterative karakter	17
2.5. Det tekniske arbejdes forløb.....	19
2.6. Arbejdsdeling og Kooperation.....	21
2.7. Opgavespektrum.....	24
2.8. Det tekniske arbejdes variation.....	26
2.9. Opsummering.....	26
3. Teknikerens Integreerede Arbejdsplads.....	28
3.1. Organisationens integrerede database.....	28
3.2. Teknikerens arbejdsstation.....	29
3.2.1. Iterationsstøtte	30
3.2.2. Funktionalitetsspektrum.....	31
3.2.3. Integration af opgaver.....	33
3.2.4. Brugergrænseflade.....	33
3.3. Understøttelse af kooperativt teknisk arbejde.....	37
3.4. Åben systemarkitektur.....	37
4. Konklusion.....	39
Litteratur.....	40

Forord

Teknikerens arbejdsplads er det system af ressourcer (maskinel, programmel, informationskilder, arkiver, faciliteter i øvrigt), der støtter teknikeren i arbejdet. Teknikerens Integrerede Arbejdsplads kan således defineres som *et datamat-integreret støttesystem for teknikere*.

Med det formål at afklare kravene til Teknikerens Integrerede Arbejdsplads, undersøge 'state of the art' inden for feltet samt afklare konceptets muligheder og begrænsninger, blev der i maj 1986 påbegyndt et projekt under navnet: *Forstudie til Teknikerens Integrerede Arbejdsplads (TIA)*. Projektet er blevet gennemført af Jysk Teknologisk og Dansk Datamatik Center i samarbejde med en række store danske industrivirksomheder, der anvender edb-systemer til understøttelse af teknikeres arbejde (Lego, Aalborg Boilers, Novo, Grundfos, Danfoss, Søren T. Lyngsøe og F.L. Smidth), en række førende leverandører af edb-systemer til sådanne formål (IBM, Hewlett Packard, Digital og Dansk Data Elektronik) samt Industrirådet og Jernets Arbejdsgiverforening. Projektet har modtaget støtte fra Teknologirådet.

En række førende udenlandske og danske forskere inden for en række forskningsfelter af central betydning for datamatbaserede tekniske arbejdspladser har været inddraget i diskussionerne: Donald O. Knight, Arizona State University; William Buxton, University of Toronto; Jens Rasmussen, Forskningscenter Risø; Jakob Nielsen, Institut for Datateknik, DTH; Lucy Suchman, Xerox PARC, samt Roxanne Hiltz og Murray Turoff, New Jersey Institute of Technology.

Den foreliggende rapport er et forsøg på at udlede kravene til Teknikerens Integrerede Arbejdsplads fra en bestemmelse af det tekniske arbejdes karakter.

Rapporten udgør kun en del af den samlede rapportering fra forprojektet Teknikerens Integrerede Arbejdsplads.

De synspunkter, der fremføres i det følgende, udspringer af arbejdet i projektet, og har været diskuteret i projektgruppen, men det bør samtidig understreges, at de fremførte synspunkter er forfatterens ansvar, og at de ikke nødvendigvis deles af de organisationer, der har deltaget i projektet.

1. Indledning

Hvorfor datamat-baserede tekniske arbejdspladser?

Begrundelsen kan helt generelt og relativt enkelt siges at være, at informationsbehandlingen udgør en flaskehals for at øge det tekniske arbejdes produktivitet.¹

En forøgelse af arbejdsproduktiviteten er utvivlsomt et vigtigt motiv. Denne begrundelse, som man hyppigt støder på i den amerikanske litteratur, er dog langt fra dækkende for de muligheder, datamat-baserede tekniske arbejdspladser rummer.

Blandt de mest væsentlige faktorer - ud over arbejdsproduktiviteten - må således anføres:²

1) Mange tekniske arbejdsopgaver, navnlig konstruktion af meget komplekse systemer så som VLSI chips, kernekraftværker etc., kunne ikke udføres, i hvert fald ikke med den fornødne kvalitet, uden ved hjælp af datamat-baserede arbejdsstationer.

2) Datamat-baserede arbejdsstationer indebærer en voldsom forøgelse af den til rådighed stående regnekraft og gør det dermed muligt at nå længere i retning af den optimale løsning på et givet problem. Flere alternativer kan identificeres og vurderes inden for den samme tid og med de samme ressourcer. Kvaliteten af det tekniske arbejde kan derved øges.

3) Udviklingsprocessens gennemløbstid fra idé til forsendelse ("lead time") kan reduceres. Derved kan nye produkter markedsføres, før markedet for dem nærmer sig mætning. På mange områder - især brancher præget af hastig teknologisk udvikling - er timingen af lanceringen af nye produkter idag ofte en vigtigere konkurrenceparameter end produkternes produktionspris.

4) Udviklingsomkostningerne kan reduceres, især lønomkostninger; dette gælder først og fremmest omkostningerne ved modifikationer af allerede eksisterende konstruktioner. Det bliver derved økonomisk muligt at imødekomme det stadigt mere pågående krav om at kunne gå ned i seriestørrelse. Virksomheden kan derved øge sit udbud af produkter og varianter. Det bliver ydermere i højere grad muligt at imødekomme kundernes individuelle behov og ønsker ved at producere efter ordre.

5) Kvaliteten kan ofte øges (e.g. mindre tolerancer).

6) Produkterne kan blive bedre dokumenterede.

Så vidt, så godt. Men hvorledes kan teknikerens arbejde støttes med datamat-baseret teknologi på en sådan måde, at disse muligheder realiseres? Hvad kræves der i denne sammenhæng af de datamat-baserede systemer, der skal anvendes? Hvorledes skal de udformes?

Når man skal bygge en bro, bør man vide en hel del om spændbeton. Men uden viden om de lokale jordbundsforhold og om den forventede trafik bliver resultatet næppe godt.

Noget tilsvarende gælder for konstruktion af datamatbaserede systemer. Uden viden om de særlige forhold, der karakteriserer anvendelsesområdet, kan resultatet ikke blive godt.

¹ L. W. Bezanson m.fl.: "Engineering Support System User Requirements", *IEEE Micro*, oktober 1985.

² Jvf. bl.a. Raphael Kaplinsky: *Automation, the technology and society*, 1984.

Når man skal specificere kravene til datamat-baserede systemer til understøttelse af teknikeres arbejde, må man bygge på en analyse af teknikeres arbejde og udlede kravene af netop dette arbejdsdomænes karakteristika.

Som led i TIA forprojektet er der derfor blevet gennemført et mindre feltstudie hos F. L. Smidth & Co. A/S i Valby. Ialt er fem ingeniører i to forskellige arbejdsområder blevet interviewet.

Den følgende analyse bygger bl.a. på en bearbejdning af disse interviews.

Analysen bygger derudover på litteraturstudier, på inviterede oplæg og i det hele taget på diskussionerne på TIA-forprojektets seminarer samt på det arbejde, der blev udført i TIA-forprojektets arbejdsgrupper.

Det har ikke inden for TIA-forprojektets rammer været muligt at gennemføre en sammenlignende undersøgelse af teknikeres arbejde i forskellige brancher. En sådan undersøgelse ville kunne give et langt mere konkret billede af teknikerens arbejde og dermed et langt mere solidt grundlag for specifikation af kravene til Teknikerens Integrerede Arbejdsplads.

Vi har dog i analysen forsøgt at abstrahere fra de særlige forhold, der karakteriserer enkelte brancher og har i stedet bestræbt os på at fremhæve de grundlæggende karakteristika ved teknisk arbejde overhovedet. Det er således vor faste overbevisning, at analysen har almen gyldighed for teknisk arbejde generelt, og at det dermed er muligt at sige noget fornuftigt om de generelle krav til datamat-baserede systemer til understøttelse af teknisk arbejde, uanset branche.

2. Teknikerens arbejde

Genstandsområdet for teknikerens arbejde er tekniske systemers - broer, integrerede kredsløb, symaskiner, drejebænke, kraftværker, datamaskiners - livscyklus, fra idé over konstruktion og idriftsætning til destruktión. Til forskel fra arbejdsopgaver af udførende og overvågende karakter består indholdet i teknikerens arbejde i alle faser af systemets livscyklus i planlægning: planlægning af systemets udformning, planlægning af dets fremstilling, planlægning af dets idriftsætning og drift, planlægning af dets vedligeholdelse samt planlægning af dets sluttelige afvikling.

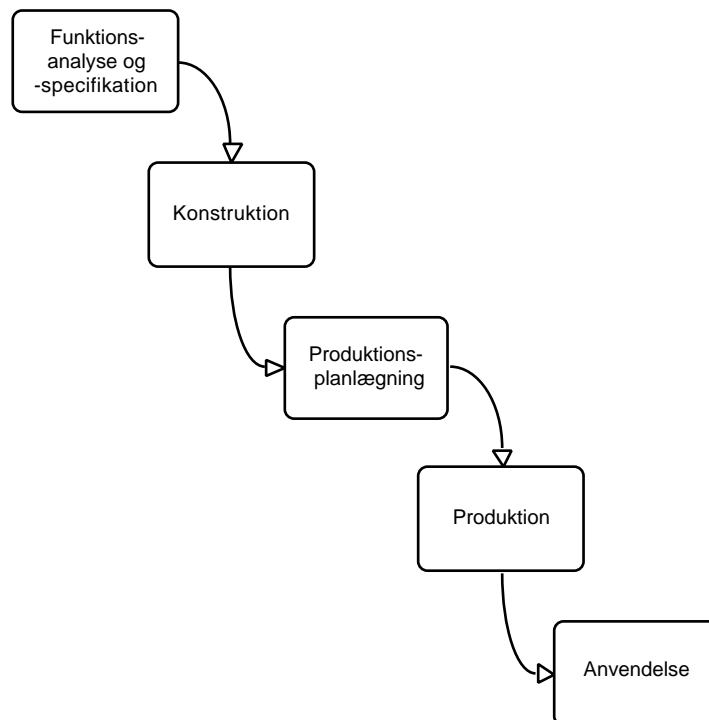


Fig. . Genstandsområdet for teknikerens arbejde er tekniske systemers livscyklus, fra idé over konstruktion og idriftsætning til destruktión.

På områder, hvor usikkerhedsmomenterne i planlægningen er mange og store, som det f.eks. ofte er tilfældet i byggebranchen, vil skellet mellem planlæggende og udførende funktioner som regel være mindre tydeligt. Teknikeren vil i sådanne tilfælde ofte forestå opgaver af udførende og overvågende karakter for at tilse, at de reelle betingelser faktisk svarer til de antagelser, der lå til grund for planlægningen, og for i påkommende tilfælde at korrigere planerne.

Eftersom de fleste beslutninger af betydning for et teknisk systems anvendelighed og pris tages eller foruddiskonteres i de første faser af systemets levetid, ved dets udvikling og konstruktion,³ beslægtet denne fase naturligt nok en uforholdsmæssig stor del af interessen. Denne fase er af tiltagende strategisk betydning for virksomhedernes evne til at klare sig.

³ Beslutninger, der fastlægger op til 80% af prisen på et produkt, træffes i forbindelse med udvikling og konstruktion. (U. Bauenfeind: "Realisierung von CIM-Konzepten mit Standardkomponenten", *Zeitschrift für wirtschaftliche Fertigung*, vol. 80, 1985, nr. 9, pp. 397-403.)

2.1. Det tekniske arbejdsdomæne

Det tekniske arbejdsdomæne befinder sig på grænsefladen mellem det menneskelige samfund og den "ydre" natur. Tekniske systemer er fysiske etc. systemer, der er frembragt af mennesker med et eller andet, samfundsmæssigt bestemt formål for øje. Tekniske systemer er altså af dobbelt natur; de er på den ene side fysiske etc. systemer, bestemt af kausale relationer ("naturlovene"), og på den anden side hensigtsmæssige systemer, skabt med henblik på at tjene et formål, der udspringer af samfundsmæssige behov.

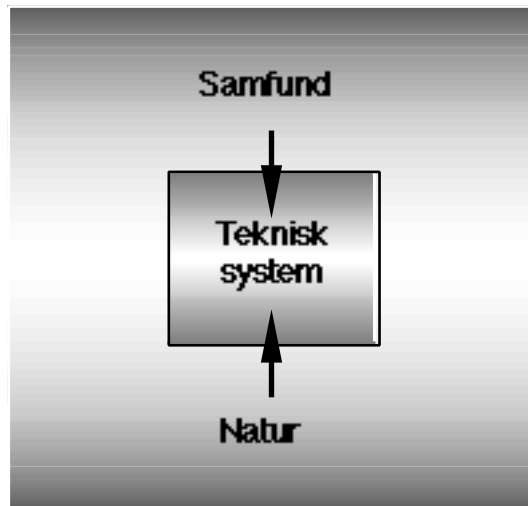


Fig.

I sit arbejde anskuer teknikeren derfor det system, han arbejder med, som en sådan enhed af et fysisk og et funktionelt system. Han anskuer det med andre ord i et mål-middel perspektiv. (Se fig. #).

Et teknisk system som f.eks. et ur er - ligeså vel som en sten - en naturgenstand. Det er underlagt og adlyder naturlove. Men samtidig er uret naturligvis udformet med henblik på opnåelse af menneskelige formål. Det er en hensigtsmæssig ting; det har en funktion. Urets formål er naivt udtrykt at "vise tiden". Urets mekanisme (tandhjul, fjeder osv.) er dets "indre omverden", der sætter det i stand til at opfylde sit formål. De specifikke betingelser, den "ydre omverden" frembyder, indebærer specifikke krav til den "indre omverden". Konstruktionen af den "indre omverden" må imødekomme disse specifikke krav, hvis uret skal være i stand til at opfylde sit formål under de specifikke betingelser, den "ydre omverden" frembyder. Et solur kan fungere udmærket i et solrigt klima, men ikke særligt godt i den arktiske vinter. Tilsvarende må den "indre omverden" hos et ur, der skal fungere under den "ydre omverden", der findes på et skib, imødekomme andre krav end f.eks. et bornholmerur.

Den specifikke omverden, hvori et system skal opfylde sit formål, bestemmer dets specifikke "indre" egenskaber. Solure, armbåndsure, bornholmerure, kronometre osv. opfylder det samme formål i forskellige "ydre omverdener" i kraft af forskellige "indre omverdener" eller mekanismer.

For at kunne opfylde sit formål skal et system kunne fungere under de betingelser, der gælder i den "ydre omverden", hvori formålet skal opfyldes.

Formålet og den “ydre omverdens” betingelser er med andre ord snævert forbundet. Man konstruerer ikke blot et ur til at “vise tiden”, men til at “vise tiden” under helt specifikke betingelser. For at udtrykke denne forbundethed anvender Simon og Newell begrebet *task environment* eller “opgave-omverden”: “Udtrykket *opgave-omverden* [...] refererer til en omverden forbundet med et mål, et problem eller en opgave.”⁴ Begrebet opgave-omverden udtrykker de aspekter ved den “ydre omverden”, der er af betydning for at opnå formålet.

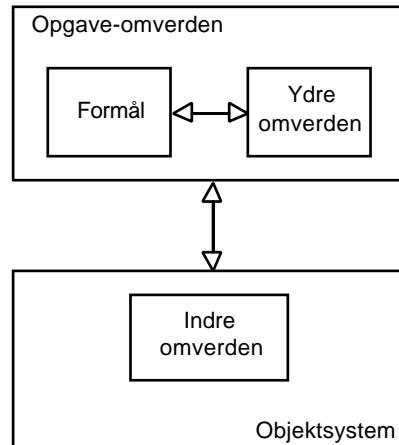


Fig. #. Et givet teknisk system, “objektsystemet”, opfylder et bestemt formål under de specifikke betingelser, den “ydre omverden” frembyder. Dets “indre omverden” (dets fysiske etc. organisering) modsvarer dets “opgave-omverden”.

På den ene side kan vi altså anskue uret som en fysisk genstand, en ansamling af tandhjul og fjedre e.l. Vi kan imidlertid også anskue uret i relation til dets formål og den specifikke omverden, hvori de skal “vise tiden”. Som Herbert Simon udtrykker det, kan et teknisk system anskues som et mødested eller “interface” mellem den “indre omverden” og den “ydre omverden”.

“An artifact can be thought of as a meeting point - an ‘interface’ in today’s terms - between an ‘inner’ environment, the substance and organization of the artifact itself, and an ‘outer’ environment, the surroundings in which it operates. If the inner environment is appropriate to the outer environment, or vice versa, the artifact will serve its intended purpose.”⁵

Designerens arbejde består således i at udforme systemets “indre omverden” på en sådan måde, at det kan opfylde sit formål under de specifikke betingelser, den “ydre omverden” frembyder.

Det tekniske arbejdes arbejdsdomæne er karakteriseret ved et hierarki af mål-middel relationer, der under ét formidler modsætningsforholdet mellem menneskelige intentioner og den “ydre” naturs modvillighed over for vore intentioner. (Se fig. #).

⁴ Allen Newell and Herbert A. Simon: *Human Problem Solving*, 1972.

⁵ Herbert A. Simon: *The Sciences of the Artificial*, 2nd ed., Cambridge, Mass.-London, 1982.

<i>Means-End Levels</i>	<i>Properties of System Selected for Representation</i>
<i>Purpose, Constraints</i>	Purpose based properties; reasons for the design are propagating top-down.
<i>Abstract Function</i>	
<i>Generalized Function</i>	Physically based properties; causes of performance are propagating bottom-up.
<i>Physical Function</i>	
<i>Physical Form</i>	

Fig. #. Mål-middel hierarki for teknisk system. (Efter Jens Rasmussen).⁶ Figuren illustrerer tekniske systemers dobbelte natur: en enhed af et intentionelt system og et kausalt system. Denne dobbelthed betyder, at et teknisk systems funktionelle egenskaber kan beskrives på flere abstraktionsniveauer.

På grundlag af en lang række empiriske analyser af de mentale repræsentationer, der anvendes i problemløsnings i relation til meget forskelligartede tekniske systemer, har Jens Rasmussen således identificeret fem niveauer:

“When moving from one level of abstraction to the next higher level, the change in system properties represented is not merely the removal of the various functions or element at the lower level. In man-made systems these higher level principles are naturally derived from the purpose of the system, i.e. from the reasons for the configurations at the level considered. Change of level of abstraction involves a shift in concepts and structure for representation as well as a change in the information suitable to characterize the state of the function or operation at the various levels of abstraction.[...] In other words, models at low levels of abstraction are related to a specific physical world that can serve several purposes. Models at higher levels of abstraction are closely related to the specific purpose that can be met by several physical arrangements.”⁷

Antallet af niveauer afhænger dog givetvis af det specifikke domænes karakter.

⁶ Jens Rasmussen: *A Cognitive Engineering Approach to the Modelling of Decision Making and its Organization in Process Control, Emergency Management, CAD/CAM, Office Systems, Library Systems*, Risø, juni 1986.

⁷ Jens Rasmussen: “The Role of Hierarchical Knowledge Representation in Decisionmaking and System Management”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-15, no. 2, March/April 1985, pp. 234-243.

<i>Means-End Levels</i>	<i>Properties of the System Selected for Representation</i>
<i>Purpose, Constraints</i>	Properties necessary and sufficient for relating the performance of the system with the reasons for design, with requirements of environment. Categorization in terms referring to properties of the environment.
<i>Abstract Function</i>	Properties necessary and sufficient to establish relationships according to design or intention; energy, value, information, truth, etc. Relationship to underlying causal structure and function is depending on convention and design choice. Categorization in abstract terms, referring neither to system nor to the environment.
<i>Generalized Function</i>	Properties necessary and sufficient to establish "black box" input-output models of functions irrespective of underlying implementation; this level is necessary for coordination of different physical processes to serve joint higher purposes. Categorization according to recurrent, familiar input-output relationships.
<i>Physical Function</i>	Properties necessary and sufficient for use of object; for adjustment of object for use, to adjust to limits of use, to predict whether objects will serve particular use to select part to move for control of physical process. Categorization according to underlying physical process.
<i>Physical Form</i>	Properties necessary and sufficient for classification and recognition of material objects.

Fig. #. Indholdet af mål-middel representation af tekniske systemer. (Efter Jens Rasmussen).⁸

2.2. Det tekniske arbejdes problemrum

Jens Rasmussen har argumenteret for, at konstruktionsarbejdet relaterer sig til tre "problemområder":

- 1) Viden om det endelige produkts funktioner og egenskaber (den ønskede tilstand).
- 2) Viden om de muligheder og begrænsninger, der karakteriserer det eksisterende produktionssystem.
- 3) Viden om det forhåndenværende udbud af eksisterende, potentielt anvendelige komponenter eller processer. (Jvf. fig. #).

Disse tre domæner udgør hver for sig et mål-middel hierarki. Tilsammen udgør disse hierarkier konstruktionsarbejdets "problemrum". Konstruktionsarbejdet udfolder sig på alle niveauer i disse tre domæner. Som Rasmussen udtrykker det:

"The design process will imply an iteration between the levels in the functional hierarchy in order to identify the acceptable functional properties of a component but, at the same time, iteration across to the manufacturing hierarchy together with consultation of the properties of the designs already in stock."⁹

⁸ Jens Rasmussen: *A Cognitive Engineering Approach to the Modelling of Decision Making and its Organization in Process Control, Emergency Management, CAD/CAM, Office Systems, Library Systems*, Risø, juni 1986.

⁹ Jens Rasmussen: *A Cognitive Engineering Approach to the Modelling of Decision Making and its Organization in Process Control, Emergency Management, CAD/CAM, Office Systems, Library Systems*, Risø, juni 1986.

Anskuet som videnbaseret arbejde, består konstruktionsarbejdet grundlæggende i at relatere og kombinere viden om disse tre problemdomæner. Eftersom der er tale om en samordningsproces, er processen langt fra lineær. Teknikeren vil mentalt bevæge sig omkring i hele problemrummet i en søgning efter en tilfredsstillende løsning.

Efterhånden som konstruktionsprocessen skrider fremad, vil der dog normalt ske en vis forskydning af vægten fra interesse for objektsystemets funktionelle egenskaber til interesse for fremstillingssystemets og de forhåndenværende konstruktioners egenskaber.

De øvrige afarter af teknisk arbejde, der knytter sig til de senere faser af tekniske systemers livscyklus, arbejder inden for det samme problemrum, selv om de dog ikke (som konstruktionsarbejdet) relaterer sig til hele problemrummet. (Der er tale om forskellige funktioner inden for samme domæne).

<i>Functional System Environment</i>	<i>Manufacturing Environment</i>	<i>Attributes of Components on Stock</i>
Functional system goals and constraints	Manufacturing goals and constraints	Functional goals and constraints of previous applications
Customer's priority setting; flow of energy, material, and monetary values, manpower etc.	Manufacturer's priority setting; flow of energy, material, and monetary values, manpower, etc.	Price, reliability and other higher order spec's
General function to be performed in target system: bearing, support, conduct, transfer	General manufacturing systems available: casting, forging, assembling, cutting, etc.	Inted functional context of initial design
Acceptable physical functions of separate part	Physical forming processes available	Physical functioning spec's
Related functional components	Part to be manufactured	Shape, size, material
Available tools and machinery		
Geometrical form, material, surface finish		

Fig. #. (Efter Jens Rasmussen).¹⁰

Til disse tre problem-domæner må føjes et fjerde, omend på et andet niveau: viden om teknikerens arbejdsplads. Teknikeren må i sit arbejde også relatere sig til de ressourcer, værktøjer etc., der er til rådighed for selve designprocessen. Hvad teknikeren overhovedet ikke kan gøre med de til rådighed stående ressourcer, vil ikke blive gjort. Og hvad han kun kan gøre med stort besvær eller forsinkelse, vil ikke blive gjort, hvis det kan undgås. Teknikerens viden om sine ressourcer som

¹⁰ Jens Rasmussen: *A Cognitive Engineering Approach to the Modelling of Decision Making and its Organization in Process Control, Emergency Management, CAD/CAM, Office Systems, Library Systems*, Risø, juni 1986.

tekniker indgår derfor i hans vidensgrundlag på linje med hans viden om de andre domæner.

2.3. Grundfunktionerne i teknikerens arbejde

Der findes mange slags teknikere. Enhver person, der besidder og anvender en faglig kunnen på løsning af tekniske planlægningsproblemer, er en tekniker. En edb-programmør, der koder og afprøver et lønsystem; en kemiker, der udvikler et nyt beroligende middel; en arkitekt, der forestår et byggeri; en ingeniør, der konstruerer og afprøver en ny pumpe, - alle er de teknikere i ordets normale forstand. Det, der er fælles for de mange forskellige afarter, er *teknisk problemløsning*.

Hvad er problemløsning? Helt alment foreligger et problem, når den eksisterende tilstand på et felt i en eller anden forstand afviger den ønskede, og når vejen til at opnå den ønskede tilstand ikke er kendt på forhånd. Problemet løses således ved, at der findes en vej til den ønskede tilstand. Teknikerens arbejde består i at finde vejen til løsning af problemer af teknisk art.

Teknisk problemløsning som domæne omfatter alment følgende funktioner:

- Formulering af problemet. (Hvori består problemet? Er der overhovedet et problem?)
- Dannelse af en model af opgaveomverdenens krav til objektsystemet.
- Nedbrydning af problemet i overkommelige delproblemer og koordination af delløsningerne.
- Konceptualisering, dvs. søgning efter viden om evt. midler (strukturer, processer, mekanismer, materialer etc.), der kan imødekomme disse krav.
- Udvælgelse af egnede kandidater blandt de evt. egnede midler, der måtte være fundet.
- Konstruktion af en eller flere konceptuelle modeller af objektsystemets "indre", dvs. fastlæggelse af objektsystemets indre struktur.
- Verifikation af de konceptuelle modellers adfærd under forventelige betingelser.
- Evaluering af resultaterne af verifikationen.
- Udvælgelse af den mest tilfredsstillende løsning.
- Dokumentation af den fundne løsning.
- Planlægning af idriftsætning.
- Evaluering af idriftsætningserfaringer med henblik på evt. ændringer i konstruktionen.
- Evaluering af driftserfaringer med henblik på evt. ændringer i konstruktionen.

Disse almene tekniske funktioner danner fire funktioner, der udgør to komplementære funktionspar, selve kernen i teknisk arbejde: på den ene side søgning efter mulige alternative løsninger og afprøvning af de således identificerede evt. egnede alternative løsninger og på den anden nedbrydning af problemet i overkommelige delproblemer og koordination af delløsningerne.

I det følgende skal disse centrale funktionspar analyseres mere indgående.

2.3.1. KONCEPTUALISERING: SØGNING EFTER MULIGE ALTERNATIVE LØSNINGER

Teknikerens arbejde består altså i at finde veje til hensigtsmæssig tilpasning af et teknisk system til dets omverden, dvs. fastlæggelse af de egenskaber ved systemets indre omverden, der kan opfylde formålet i den givne ydre omverden. Teknikeren søger efter egnede midler til et bestemt formål. Alternative mulige løsninger vurderes. Blandt de mulige løsninger udsøges og vælges netop dén mulige løsning, der opfylder de betingelser, den "ydre omverden" frembyder, og som samtidig maksimerer nytte-funktionen.

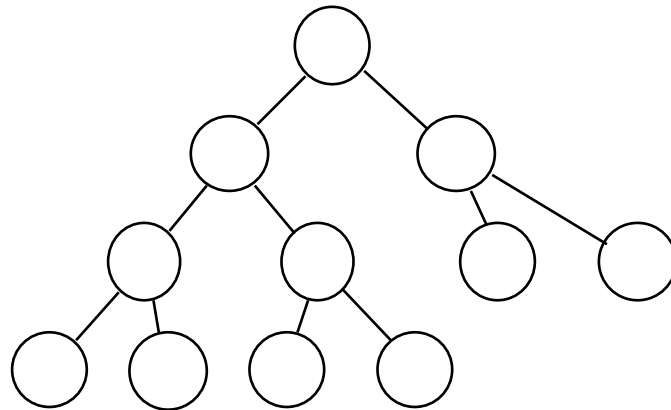


Fig. #.

Under normale omstændigheder er det imidlertid umuligt at finde den optimale løsning. Vi kan normalt ikke i praksis identificere alle tilladelige alternativer og bestemme deres respektive fortrin; dertil er den til rådighed værende regnekraft alt for begrænset. Vi kan ej heller identificere det bedste alternativ, før vi kender dem alle. Målet for søgeprocessen er derfor ikke den bedste af alle verdener, men blot en tilfredsstillende løsning.¹¹

2.3.2. VERIFIKATION AF EVT. EGNEDE ALTERNATIVE LØSNINGER

Teknikerens arbejde bevæger sig altså grundlæggende i en mål-middel dimension. Arbejdet består helt alment i at søge efter og identificere midler, der kan opfylde et bestemt formål under bestemte funktions- og produktionsbetingelse. Denne søgeproces er imidlertid uhyre problematisk.

Teknisk arbejde relaterer sig i allerhøjeste grad til "den virkelige verden". For det første er arbejdsdomænet overordentlig komplekst, og for det andet er den viden, der ligger til grund for beslutningsprocessen, principielt ufuldstændig.

*"A complete unitary description of the demands made by the context [...] is impossible. [...] What does make design a problem in real world cases is that we are trying to make a diagram for forces whose field we do not understand."*¹²

Under problemløsningsprocessen vil teknikeren derfor ustandseligt erfare, at de oprindelige antagelser ikke er holdbare. Han vil kunne erfare, at forståelsen af problemets karakter ikke var korrekt; at opgaveomverdenen stiller andre krav, end

¹¹ Herbert A. Simon: *The Sciences of the Artificial*, 1981, pp. 138-140.

¹² Christopher Alexander: *Notes on the Synthesis of Form*, 1964, p. 21.

først antaget; at den konciperede løsning ikke lever op til opgaveomverdenens krav, etc.

Af denne grund har teknisk arbejde alment udpræget *iterativ* karakter.¹³

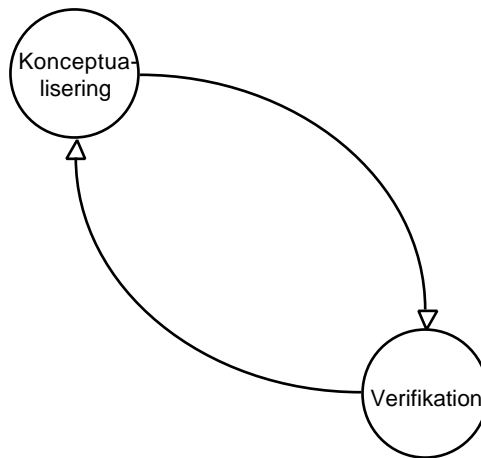


Fig. #. Teknisk arbejde udføres på grundlag af principielt ufuldstændig viden. Teknisk problemløsning er altså erfaringsbaseret arbejde og har derfor udpræget iterativ karakter. Teknisk arbejde er med andre ord kendetegnet ved en dialektik mellem *konceptualisering* (søgning efter evt. egnede midler) og *verifikation* (afprøvning).

2.3.3. NEDBRYDNING OG KOORDINATION

Teknikerens arbejde udspilles som vist i et komplekst problemrum, og det beror ydermere principielt på ufuldstændig viden, hvorfor det har en udpræget iterativ karakter. Men selv denne karakteristik giver et forenklet billede af det tekniske arbejde.

De forskellige krav, opgaveomverdenen stiller til et teknisk system, er ikke indbyrdes uafhængige. Tværtimod. De spiller sammen på mangfoldig vis. Ved konstruktion af f.eks. en støvsuger må man tage hensyn til, at jo færre forskellige materialer, der anvendes, desto mere effektiv kan montagen tilrettelægges. Dette krav om enkelhed er imidlertid i modstrid med, at systemet vil fungere bedre, hvis man anvender det bedste materiale til hvert enkelt formål. På den anden side vil mangfoldighed af materialer betyde, at det er dyrt og besværligt at sammenføje mellem komponenterne. Kravene til enkelhed, ydedygtighed og sammenføjning er tillige i modstrid med ønsket om lave omkostninger. Hvis vi anvender det billigste materiale til hvert enkelt formål, vil vi ikke nødvendigvis opnå enkelhed, ønskelig ydedygtighed eller materialer, der let lader sig samle. Situationen er typisk for teknisk problemløsning. Der er krav, der skal opfyldes, og der er vekselvirkning mellem kravene, som gør det vanskeligt at opfylde kravene. (Jvf. fig. #).

¹³ G. Bechmann, R. Vahrenkamp und B. Wingert: *Mechanisierung geistiger Arbeit. Eine sozialwissenschaftliche Begleituntersuchung zum Rechnereinsatz in der Konstruktion*, 1979.

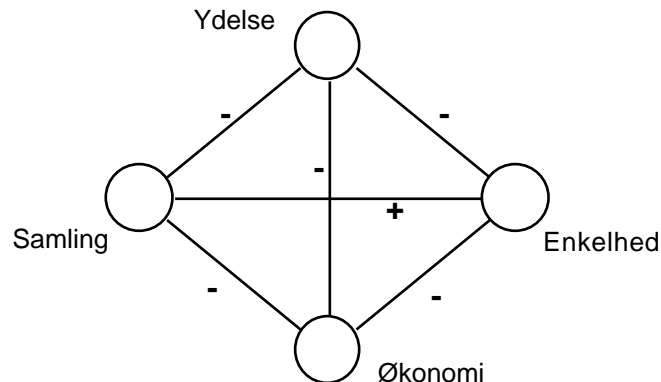


Fig. #. Eksempel på vekselvirkning mellem krav til et - simpelt - teknisk system. En forbindelse markeret med et minus repræsenterer en modsætning mellem de modstående krav, mens en forbindelse markeret med et plus repræsenterer en positiv overensstemmelse mellem de modstående krav. (Efter Christopher Alexander).¹⁴

I dette eksempel var antallet af vekselvirkninger overskueligt. Problemet er derfor simpelt, og kan forholdsvis let løse af et enkelt individ. Men hvis antallet af faktorer og vekselvirkninger mellem dem er stort, melder vanskelighederne sig for alvor.

Teknikeren vil derfor normalt “opløse” eller “nedbryde” problemet i overskuelige og overkommelige delproblemer. Fremgangsmåden ved problemløsning i forhold til et komplekst system består således i at opdage, hvorledes man kan nedbryde systemet i relativt uafhængige funktionelle subsystemer.¹⁵ Pointen er, at identificere ansamlinger af intens vekselvirkning og definere disse ansamlinger som subsystemer. Vekselvirkningen vil således være større inden for disse subsystemer end mellem dem. Ved konstruktion af et automobil vil man således behandle konstruktionen af chassis, motor, transmission, det elektriske system etc. som relativt selvstændige subsystemer med høj grad af indre vekselvirkning og med betydeligt mindre grad vekselvirkning mellem subsystemerne. Ved en sådan nedbrydning kan antallet af faktorer, der på et givet punkt i beslutningsprocessen skal tages i betragtning, dvs. søgerummets størrelse, reduceres betragteligt. Hvis subsystemerne hver for sig har et overskueligt omfang, kan de således angribes hver for sig, som var der tale om relativt simple systemer.

Proceduren er logisk, men kræver mere end logik, nemlig viden:

“if an engineer has no store of bits of background information, no matter how hard he tries to set the stage, nor how long he waits for perception to strike, no answer will appear.”¹⁶

Ved denne nedbrydning (“decomposition”) bygger teknikeren desuden på sin viden om produktionsapparatets kapacitet (hvad er overhovedet muligt? osv.) og på sin viden om forhåndenværende, potentielt egnede komponenter.

Nedbrydning af et teknisk problem i et hierarki af delproblemer og konstruktion af et modsvarende hierarki af subsystemer gør det desuden afgørende billigere at producere komplekse tekniske systemer som f.eks. produktionsanalæg eller

¹⁴ Alexander: *Notes on the Synthesis of Form*, 1964, p. 2.

¹⁵ Jvf. Alexander: *Notes on the Synthesis of Form*, 1964; Simon: *The Sciences of the Artificial*, 1981, og G. Bechmann, R. Vahrenkamp und B. Wingert: *Mechanisierung geistiger Arbeit. Eine sozialwissenschaftliche Begleituntersuchung zum Rechnereinsatz in der Konstruktion*, 1979, pp. 165 f.

¹⁶ Joseph Harrington: *Understanding the Manufacturing Process. Key to Successful CAD/CAM Implementation*, 1984.

integrerede kredsløb i små serier eller efter ordre. Omkostningerne ved udvikling af et bestemt subsystem kan således fordeles på alle de anlæg, hvori subsystemet udgør en komponent.

Nedbrydning af et problem til delproblemer har imidlertid sine omkostninger. I det omfang, systemet nedbrydes i subsystemer, opstår der behov for at koordinere deløsningerne. Det vil f.eks. være påkrævet løbende at overveje, hvorledes en ændring af et enkelt subsystem influerer på systemets øvrige subsystemer og på det totale systems præstation. Nedbrydning og koordination er altså komplementære funktioner.

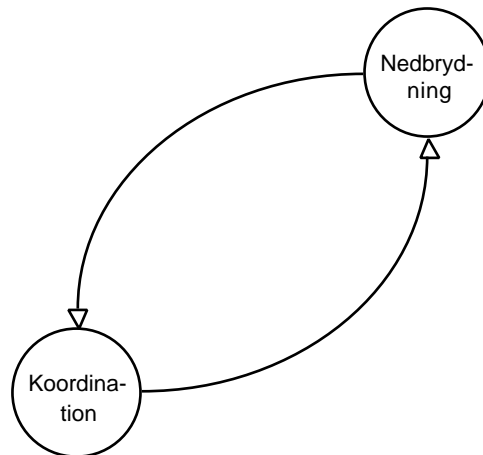


Fig. #. Teknisk arbejde er karakteriseret ved en dialektik mellem *nedbrydning* af problemet i mere overkommelige delproblemer og *koordination* af deløsningerne.

Ved komplekse systemer vil det i praksis være umuligt at sikre, at beslutninger til opfyldelse af krav angående ét subsystem ikke har utilsigtede, men ikke umiddelbart indlysende vekselvirkninger med krav til elementer i et andet subsystem. Kun gennem en iterativ proces med koncipering og verifikation er det muligt i tilfredsstillende omfang at sikre, at de forskellige subsystemer er koordineret. Ved teknisk arbejde, der består i konfiguration af komplekse tekniske systemer som f.eks. produktionsanlæg eller integrerede kredsløb på grundlag af et forhåndenværende sortiment af færdige subsystemer, og som for så vidt ikke er nykonstruktion, antager arbejdet altså igen en udpræget iterativ karakter.

Det tekniske arbejdes iterative karakter udspringer altså for det første af, at *videngrundlaget principielt er ufuldstændigt*, og for det andet af, at *søgerummet* (antallet af krav og vekselvirkninger mellem dem) *er uoverkommeligt stort*.

Det tekniske arbejde er med andre ord karakteriseret ved en dobbelt iterativ bevægelse bestemt af dialektikken mellem *konceptualisering* søgning og *verifikation* og dialektikken mellem *nedbrydning* og *koordination*.

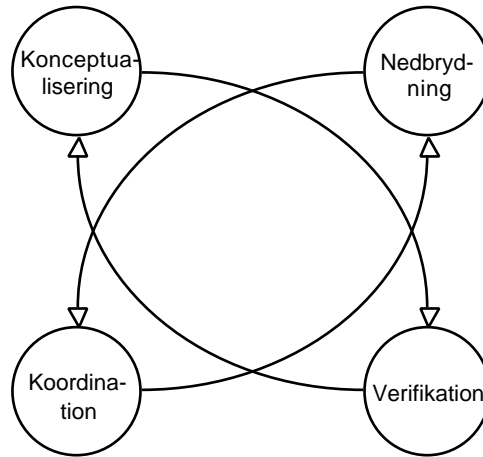


Fig. #. Det tekniske arbejde er karakteriseret ved en dobbelt iterativ bevægelse bestemt af dialektikken mellem *konceptualisering* og *verifikation* og dialektikken mellem *nedbrydning* og *koordination*.

2.4. Det tekniske arbejdes iterative karakter

I virkeligheden kan tekniske systemers hele livscyklus anskues som en gevaldig iterativ erfaringsproces, med et utal af underordnede iterationer i et utal af led. (Jvf. fig. #). Dialektikken konceptualisering-verifikation og nedbrydning-koordination karakteriserer både processen som helhed og de forskellige underordnede delprocesser.

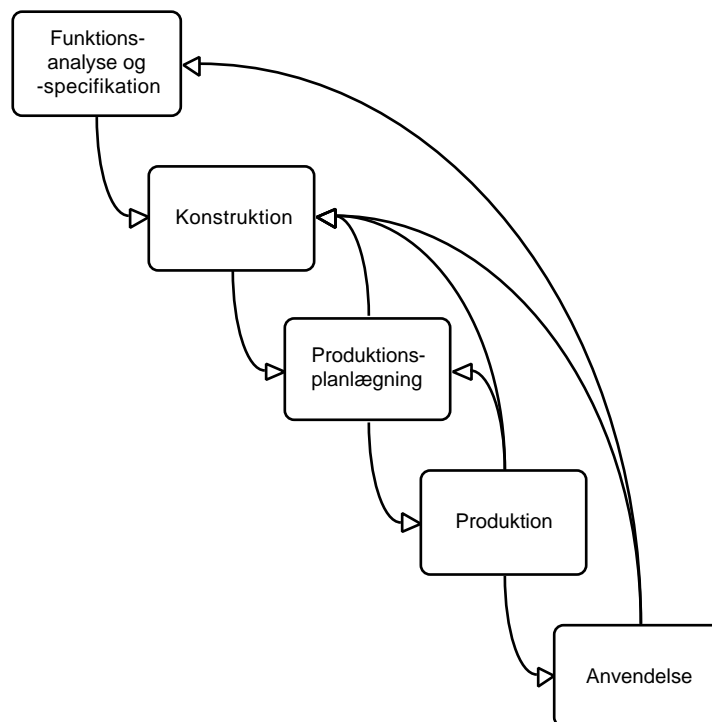


Fig. #. Figuren illustrerer den iterative karakter af arbejdet i forhold til tekniske systemers livscyklus og dermed den iterative karakter af teknisk arbejde overhovedet.

De overordnede krav til teknikerens arbejde kan sammenfattes i ordene: pålidelighed, forudsigelighed og dokumenterbarhed. For at sikre holdbare og levedygtige løsninger på problemerne, forsøger teknikeren systematisk at gøre sådanne erfaringer. De udviklede koncepter må altså så vidt muligt afprøves bevidst og systematisk under kontrollerede og realistiske betingelser.

Det er nødvendigt løbende at afprøve, om den konceptuelle model af det tekniske system svarer til de krav, opgaveomverdenen stiller til det. Den mest givende måde er naturligvis at konstruere systemet og afprøve det under reelle betingelser. Allerede idriftsatte systemer leverer således et væld af erfaringer med henblik på konstruktion af nye systemer. Men som *metode* betragtet er fremgangsmåden dog uheldig. De således konstruerede systemer vil næppe være pålidelige og forudsigelige. Af sikkerhedsmæssige, markedsføringsmæssige, etiske o.l. grunde er det derfor som regel uholdbart at *satse* på denne form for afprøvning.

Af de samme grunde kan også konstruktion og afprøvning af en *prototype* af det konciperede system være uacceptabel. Under sådanne omstændigheder vil man - hvor det er muligt - forsøge at simulere modellens adfærd under de forventede funktionsbetingelser. Men også hvor konstruktion og afprøvning af en prototype måtte være acceptabel, vil omkostningerne ved at fremstille og afprøve prototyper variere kraftigt. Og det kan i sådanne tilfælde være forbundet med så store omkostninger (sammenlignet med den forventede indtjening), at man viger tilbage herfor. Ved f.eks. udvikling af edb-programmer er omkostningen ved en iteration normalt forsvindende, mens omkostningerne ved et tilsvarende afprøvningsforløb ved udvikling af nye systemer til cementfabrikation er meget store. "Rapid prototyping" af en cementovn med en kapacitet på 3000 tons per døgn er ikke nogen realistisk fremgangsmåde. I stedet vil man - hvor det er muligt - forsøge at *simulere* modellens adfærd under de forventede funktionsbetingelser. Jo større omkostningerne ved gennemførelse af afprøvninger er, desto større betydning har derfor simulation af det planlagte systems adfærd i udviklingsarbejdet.

Men heller ikke simulation er muligt under alle omstændigheder, f.eks. ved udformning af boligbyggeri. I sådanne tilfælde må man evt. nøjes med at fremlægge den konceptuelle model til *debat* blandt interesserede parter.

Som oftest vil man ved valideringen og verifikationen af den konceptuelle model betjene sig af en kombination af disse fremgangsmåder.

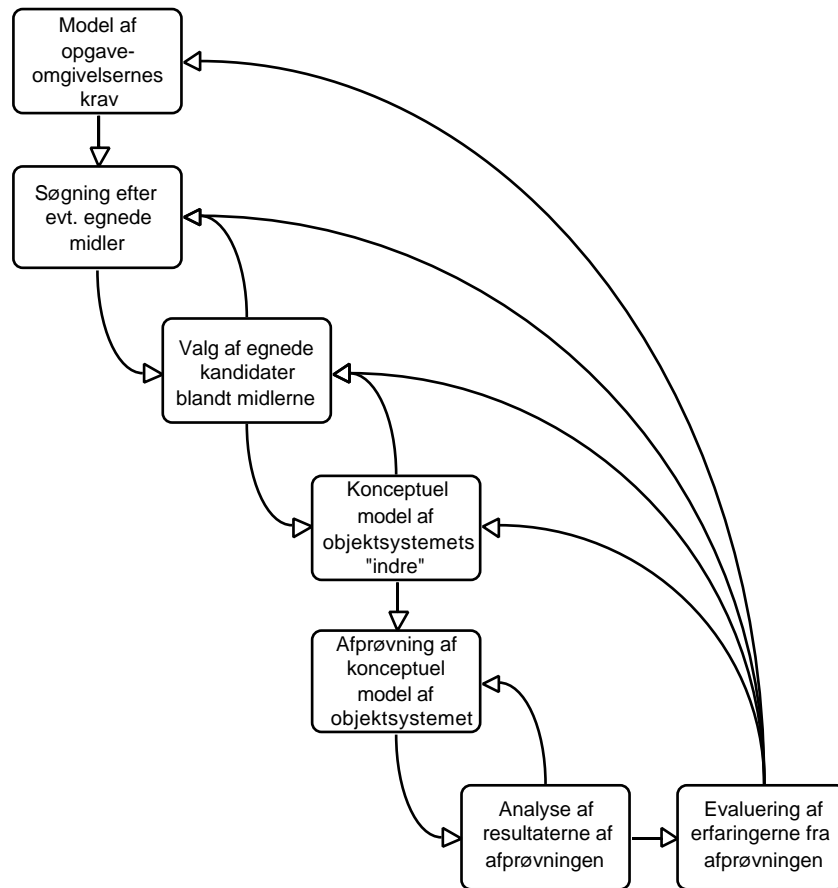


Fig. #. Eftersom teknisk arbejde er erfaringsbaseret, er det afgørende for konstruktionsprocessen at afprøve holdbarheden af de identificerede evt. egnede midler i almindelighed og den konceptuelle model i særdeleshed. Konstruktionsprocessen er således i sig selv udpræget iterativ.

2.5. Det tekniske arbejdes forløb

Selv om teknisk arbejde er kendetegnet ved en dobbelt iterativ bevægelse, er det dog som regel ikke en march på stedet. Arbejdet skrider som regel fremad. Denne bevægelse fremad forener dialektikken konceptualisering-verifikation og nedbrydning-koordination.

Der findes flere bud på en opdeling af det tekniske arbejdes fremadskridende bevægelse. Hvad selve udviklingsfunktionen angår, skelner f.eks. Harrington¹⁷ mellem tre hovedtrin:

1) *Konceptuel konstruktion*. I første række fastlægges det overordnede koncept, på grundlag af viden om objektsystemets tilsigtede opgaveomverden.

“This function is more closely focused on the technology of the company’s product than on the technology of producing that product. For example, if a company is making spinning frames for the textile industry, its research and conceptual design might be most concerned with new

¹⁷ Jvf. Joseph Harrington: *Understanding the Manufacturing Process. Key to Successful CAD/CAM Implementation*, 1984. Jvf. for en næsten identisk opdeling G. Bechmann, R. Vahrenkamp und B. Wingert: *Mechanisierung geistiger Arbeit. Eine sozialwissenschaftliche Begleituntersuchung zum Rechnereinsatz in der Konstruktion*, 1979, og Raphael Kaplinsky: *Automation, the technology and society*, 1984.

textile fibers and their behavior, and on optimum doffing cycles to achieve maximum throughput with minimum operator attention, and similar problems. The problems of designing very high speed bobbin spindles would be left to later stages of the development function.”

2) *Foreløbig konstruktion*. Dernæst identificeres og afprøves de funktionelle subsystemer.

“[...] if the product is, for example, a complete machine of a new sort or a product of similar complexity, then the preliminary design stage may be lengthy. [...] Formulating a preliminary design is one variety of problem solving. The desired end is prescribed. The available knowledge is at hand. The task is to find the bridge between them. The procedure is logical: You attack the problem by all known analytical methods; you subdivide the problem into subproblems and sub-subproblems, and analyze each.”

Konceptet vil blive udbygget til et komplet system. Der vil evt. blive bygget prototyper af de forskellige subsystemer, som afprøves. Dernæst afprøves, hvorledes subsystemerne virker sammen.

“It is almost certain, except in the simplest of cases, that the preliminary design will go through a number of iterations of the foregoing process. There will be disappointments and failures to be corrected. It will be apparent that there is a better configuration than the original best guess at configuration. Tests will be miserable failures, and changes will be required. The whole preliminary design process, or parts of it, may be reiterated many times.”

Til forskel fra konceptuelt design må foreløbigt design beskæftige sig indgående med det foreliggende produktionssystemets kapacitet:

“An important consideration in the preliminary design stage is the producibility of the design in the company’s facilities and with their available technology. [...] close feedback from Production to Design is particularly important in fields where production technology is developing almost as fast as design technology.”

2) *Detailkonstruktion*. Først når den konceptuelle model af objektsystemet som helhed og de konceptuelle modeller af subsystemer på alle niveauer er udarbejdet, afprøvet og accepteret, udarbejdes produktionsorienterede specifikationer af samtlige subsystemer.

“This final stage of the development process represents a not inconsiderable expenditure of time and effort, and the step is not taken lightly.”

Ved mekanisk produktion fastlægges først de forskellige komponenters egenskaber. Dette kan f.eks. kræve, at teknikeren gennemfører en række beregninger af alle vigtige komponenter for at afpasse dimensioner, materialer og fremstillingsteknik indtil hver komponent skønnes at være i stand til at opfylde sin funktion. Dernæst udarbejdes produktionstegninger af de enkelte komponenter. Der stilles meget store krav til disses fuldstændighed og entydighed. De er kostbare at fremstille, og det er derfor særdeles hensigtsmæssigt undgå at tegne den samme komponent to gange (“gruppeteknologi”). Næste trin er at kontrollere det forudgående arbejde ved at udarbejde montage-tegninger. Endelig udarbejdes styklistes.

Disse tre trin følger logisk efter hinanden, men ikke strengt kronologisk. Erfaringer, der gøres på et senere trin, kan føre til revision af koncepter etc., udformet på et tidligere trin og således til en iteration af det tidligere trin. Jvf. fig. #, der gengiver Harringtons model af udviklingsfunktionen. Det tekniske arbejdes iterative karakter trænger igennem overalt.

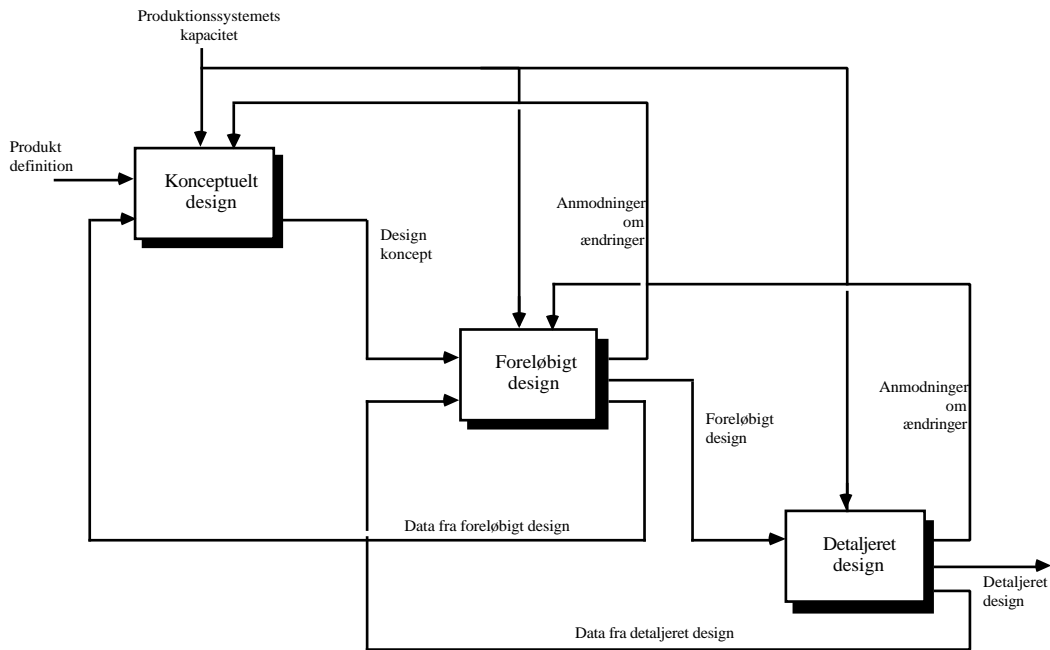


Fig. #. Udviklingsfunktionens struktur, forenklet model. IDEF-notation. (Efter Harrington).

2.6. Arbejdsdeling og Kooperation

Teknisk arbejde har udpræget kooperativ karakter. Der er tale om “distribueret problemløsning”. Der er derfor tale om et arbejdsdomæne karakteriseret ved en meget høj frekvens af kommunikation mellem beslutningstagerne.

Arbejdets kooperative karakter har forskellige årsager, og svarende hertil har kooperationen forskelligt indhold.

Kooperation kan for det første være begrundet i den omstændighed, at den informationsmængde, der skal behandles, overstiger ét menneskes informationsbehandlingskapacitet, og at behandlingsarbejdet derfor må fordeles på flere personer. De koopererende bidrager i så fald med i det væsentlige identiske faglige kvalifikationer, men varetager i den givne situation forskellige delopgaver i forbindelse med løsning af én og samme opgave. De arbejder med andre ord i det væsentlige parallelt.

Denne form for Kooperation er altså afhængig af den informationsbehandlingskapacitet, der står til rådighed på ethvert givet punkt i beslutningsprocessen. Jo mere informationsbehandlingskapacitet, den enkelte råder over, desto mindre er behovet for Kooperation om opgavens løsning. Den forøgelse af den til rådighed stående informationsbehandlingskapacitet, som datamatstøttet teknisk problemløsning indebærer, reducerer altså omfanget af Kooperationen ved løsning af en given opgave.

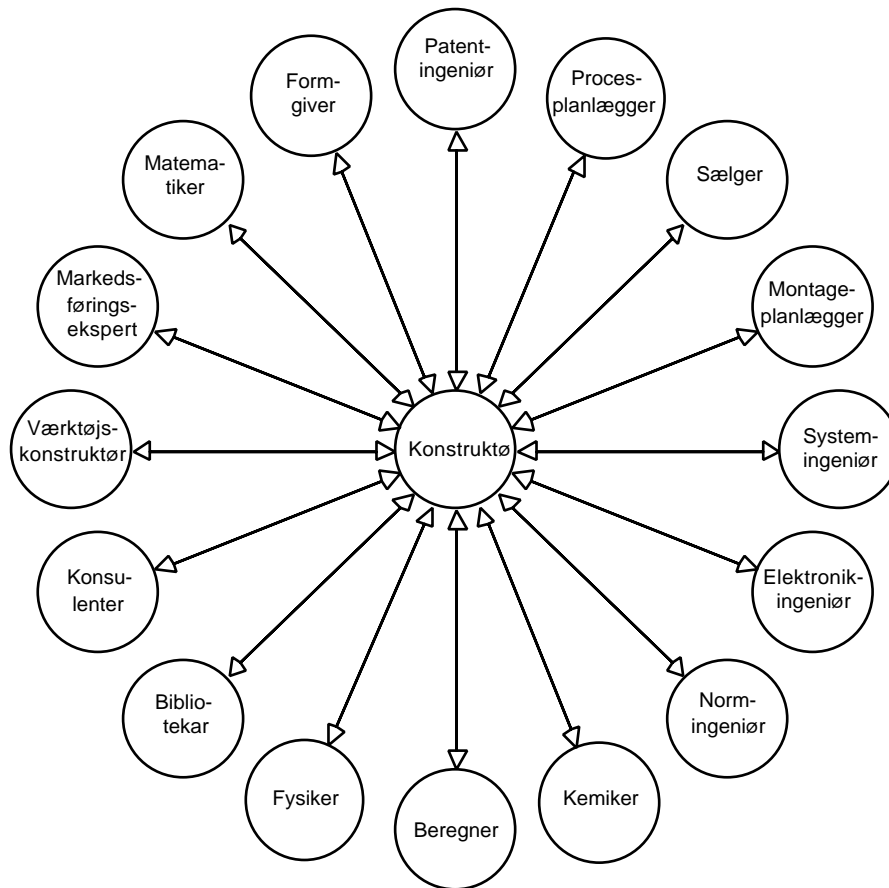


Fig. #. Teknisk arbejde er en udpræget kooperativ foreteelse. Tegningen illustrerer de kategorier af teknikere, med hvem en konstruktør samarbejder. (Efter Bechmann et al.)¹⁸

Kooperation kan for det andet være begrundet i behovet for faglig specialisering af de deltagende beslutningstagere. I dette tilfælde bidrager de koopererende med kvalitativt forskellige faglige kvalifikationer til løsning af én og samme opgave. Eftersom det tekniske arbejdsdomæne er så komplekst, som tilfældet er, indebærer teknisk problemløsning som regel et samvirke af et stort antal fagdiscipliner.

Specialiseringen er på sin side struktureret på grundlag af flere faktorer:

- 1) Teknisk arbejde er videnskabeligt baseret arbejde. Opdelingen af den videnskabelige verden i en række videnskabelige discipliner og fagområder afspejler sig naturligt nok - ikke mindst i kraft af de tekniske uddannelsers opbygning - i struktureringen af arbejdsdelingen i det tekniske arbejdsdomæne.
- 2) Faserne i systemernes livscyklus. Som regel falder de forskellige stadier i produktets livscyklus (konceptuel konstruktion, foreløbig konstruktion, detailkonstruktion, produktionsplanlægning, procesplanlægning etc.) som relativt selvstændige iterative transformationer, der stiller særlige faglige krav.

¹⁸ G. Bechmann, R. Vahrenkamp und B. Wingert: *Mechanisierung geistiger Arbeit. Eine sozialwissenschaftliche Begleituntersuchung zum Rechnereinsatz in der Konstruktion*, 1979, p. 168.

- 3) Systemernes kompleksitet. Ved konstruktion af komplekse systemer vil arbejdsdelingen ofte følge nedbrydningen af problemet i delproblemer. De forskellige typer af subsystemer konstrueres af specialiserede grupper.
- 4) Opgavernes indhold af skønsmæssig beslutningstagen. En række delopgaver, der ikke direkte relaterer sig til teknisk problemløsning, men som snarere har karakter af hjælpeopgaver til de centrale tekniske beslutningsprocesser (tegning, beregning, regnskab, tekstbehandling, grafisk illustration etc.), kan i nogen grad med fordel udskilles og varetages af andre medarbejderkategorier, typisk medarbejdere uden videregående teknisk uddannelse.

Også denne form for Kooperation er afhængig af arbejdets teknologiske grundlag. Udviklingen af datamatstøttet teknisk problemløsning er ensbetydende med en ændring af funktionsfordelingen mellem menneske og maskine, idet denne udvikling betyder, at der udvikles maskinelle systemer, der er i stand til at udføre funktioner, der tidligere blev udført af mennesker. Eftersom funktionsfordelingen mellem menneske og maskine ændres, ændres tillige kvalifikationskravene til medarbejderne og dermed grundlaget for Kooperationen. Mere specifikt indebærer den ændrede funktionsfordeling mellem menneske og maskine, at den enkelte medarbejder - i kraft af det maskinelle systems udvidede funktionalitet - er i stand til at dække et bredere fagligt felt. Datamatstøttet teknisk problemløsning indebærer derfor tendentielt, at specialiseringen begynder at vige, og reducerer således omfanget af Kooperationen ved løsning af en given opgave.

Og endelig kan Kooperation for det tredje være begrundet i behovet for at sikre, at problemet ansues fra flere sider, og at den enkeltes vurderinger konfronteres med andre vurderinger. De koopererende bidrager også i dette tilfælde med - i det væsentlige - identiske faglige kvalifikationer, men i dette tilfælde er også deres individuelle bidrag til opgavens løsning i det væsentlige identiske.

Den sidstnævnte kooperationsform synes faktisk at være af stor betydning. Selve kernen i konstruktionsarbejdet - generering af idéer til løsning af de givne opgaver - er en fundamentalt social aktivitet. Teknikere er således særdeles afhængige af udveksling af viden og synspunkter med kolleger. En interessant undersøgelse viser således, at kvaliteten af designeres arbejde korrelerer med en høj grad af kommunikation med kolleger *uden for* den givne projektgruppe. (Jvf. fig #).

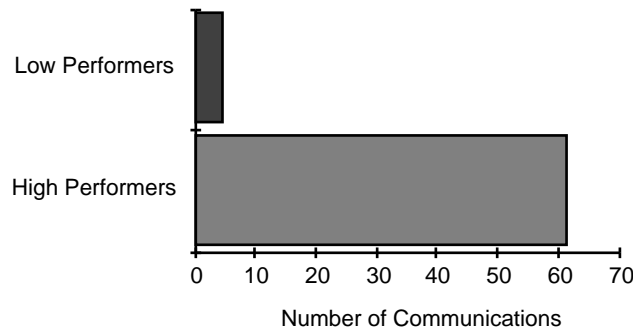


Fig. #. Hyppigheden af konstruktørers kommunikation med andre konstruktører *uden for* den givne projektgruppe, fordelt på to kategorier af konstruktører, defineret ved kvaliteten af deres arbejde: "High Performers" og "Low Performers". (Efter Thomas Allen).¹⁹

Hvad indholdet af kommunikationen, anfører Allen blot:

"Project teams are absolutely dependent upon technical information from beyond their membership, and consultation with laboratory colleagues is the most effective way of fulfilling this need."²⁰

Men "teknisk information" er jo et vidt begreb. I betragtning af, at der netop er tale om kommunikation med kolleger uden for den givne projektgruppe, er det næppe primært faktisk viden, der formidles, men nok snarere vurderinger, skøn, modspil. Det er f.eks. tænkeligt, at årsagen til det fænomen, Allen beskriver, er at teknikeren gennem meningsudveksling med fagfæller uden for projektgruppen skærper følsomheden over for evt. problemer og mangler ved det koncept, han eller hun p.t. arbejder med. Netop fordi kollegerne er uden for projektgruppen, vil de lettere kunne anskue problemet fra andre synsvinkler; de vil typisk benytte andre løsningsstrategier og kan dermed bidrage til at finde gode løsninger. At kvaliteten af konstruktionsarbejdet er afhængig af intens kommunikation med fagfæller uden for den givne projektgruppe, kan altså skyldes, at der er tale om problemløsningsarbejde på et principielt ufuldstændigt informationsgrundlag. Det udbedrer ikke det ufuldstændige informationsgrundlag at konsultere udenforstående kolleger, men det faktisk forhåndenværende informationsgrundlag kan udnyttes bedre, når det anskues ud fra forskellige synsvinkler.

2.7. Opgavespektrum

Arbejdsdelingen til trods er den enkelte teknikers individuelle delarbejde allerede i dag overordentlig sammensat i den forstand, at det omfatter som regel et stort antal meget forskelligartede opgaver. De decideret tekniske opgaver, han er uddannet til at varetage, beslaglægger ofte kun en ringe del af hans tid. Undersøgelser har f.eks. vist, at design og analyse kun udgør mellem 15 og 25 % af en ingeniørs arbejdsdag.²¹ Den typiske tidsfordeling ser iflg. en af disse undersøgelser således ud:

¹⁹ Thomas J. Allen: *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization*, 1977.

²⁰ Thomas J. Allen: *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization*, 1977.

²¹ L. W. Bezanson m.fl.: "Engineering Support System User Requirements", *IEEE Micro*, oktober 1985.

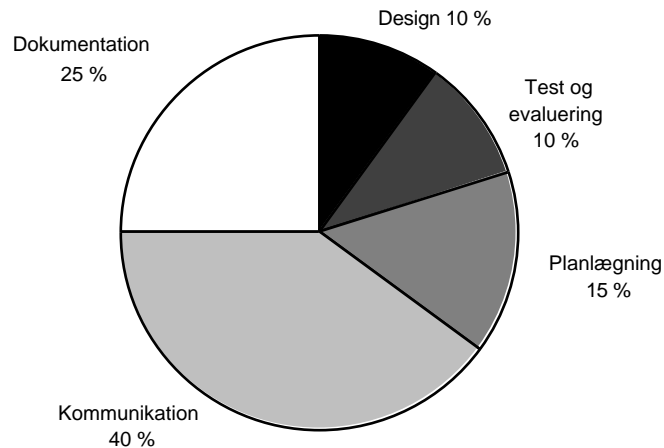


Fig. #. Eksempel på fordelingen af konstruktørers tidsforbrug på forskellige typer af opgaver. (Efter Bezanson).²²

Der er imidlertid grund til at tage resultaterne af disse tidsfordelingsanalyser meget varsomt. Man kan f.eks. ikke af den omstændighed, at kommunikation beslaglægger 40% af en teknikers daglige arbejdstid, slutte, at en potentielt meget stor rationaliseringsgevinst er at hente her. Den empirisk iagttagelige tidsfordeling siger ikke noget om, hvorfor det forholder sig således. Som anført kan den store andel af kommunikation f.eks. tænkes at være et udslag af teknikerens afhængighed af udveksling af vurderinger, og for så vidt vil denne aktivitet næppe kunne nedbringes mærkbart. Omvendt er det f.eks. tænkeligt, at kommunikationen er et udslag af den form for arbejdsdeling, der er fremherskende. En arbejdsorganisation præget af vidtgående opsplitning af konstruktionsarbejdet i delopgaver, der er allokeret til specialister, vil være præget af omfattende kommunikation, idet den enkelte konstruktionsopgave vandrer fra medarbejder til medarbejder. Beslutningsstøttesystemer, der udvider den enkeltes arbejdsfelt, vil kunne mindske opsplitningen og dermed reducere behovet for kommunikation.

De i disse undersøgelser anvendte kategorier er i det hele taget ikke uafhængige af hinanden. De faciliteter, den enkelte tekniker har til rådighed, bestemmer graden af opsplitning af arbejdet i individuelle delopgaver og dermed behovet for kommunikation mellem delarbejderne. Tilsvarende kan kraftigere redskaber til understøttelse af den egentlige konstruktionsproces (beslutningsstøttesystemer, simulering, CAD/CAE osv.) f.eks. bevirke en reduktion af af den tid, der anvendes på test og evaluering såvel som dokumentation.

Endelig er kategorierne ikke jævnbrydige. De centrale opgaver for designeren er design, test og evaluering, mens dokumentation, kommunikation og planlægning i det væsentlige er at betragte som hjælpeopgaver. Hvis den tid, teknikeren må anvende på disse hjælpeopgaver alt i alt kan nedbringes med f.eks. 20 pct.point til fordel for design, test og evaluering, vil det således betyde en fordobling af den effektive arbejdsproduktivitet.

Undersøgelserne kan med andre ord ikke direkte anvendes til at udlede kravene til TIA. (F.eks. kan man ikke sige, at beslutningsstøttesystemer til konstruktør næppe kan forventes, fordi selve konstruktionsopgaverne kun beslaglægger 10% af arbejdstiden.)

²² L. W. Bezanson m.fl.: "Engineering Support System User Requirements", *IEEE Micro*, oktober 1985.

Hvad undersøgelserne derimod viser med al ønskelig tydelighed, er, at den enkelte teknikers arbejde er meget sammensat. Dette stiller store krav til Teknikerens Integrerede Arbejdsplads i henseende til funktionalitet såvel som i henseende til integrationen af delopgaverne.

2.8. Det tekniske arbejdes variation

Vi har i det ovenstående forsøgt at fremdrage det, der generelt karakteriserer teknisk arbejde: det tekniske problemrums komplekse struktur; det tekniske arbejdes iterative karakter; det arbejdes kooperative karakter; det brede opgavespektrum, der karakteriserer den enkelte teknikers arbejde osv.

Disse væsentlige fællestræk kan dog ikke skjule den kendsgerning, at teknisk arbejde samtidig er overordentlig forskelligt. Der er for det første forskellen mellem de forskellige brancher, defineret ved produktets art, og dermed de videnskabelige discipliner, der har den centrale placering i domænet (maskinindustri, elektronisk industri, kemisk industri, plastikindustri etc.). Der er for det andet forskellen i arbejdets karakter i de forskellige faser i produktets livscyklus. Der er for det tredje forskellen mellem nykonstruktion med henblik på masseproduktion og tilpasningskonstruktion eller tilbudskonstruktion med henblik på småserieproduktion eller produktion efter ordre. Osv. Alle disse specifikke omstændigheder - og mange, mange flere - må tages i betragtning ved specifikation af kravene til Teknikerens Integrerede Arbejdsplads.

2.9. Opsummering

Sammenfattende kan designarbejdet og dets informationsbehov anskues som i fig. #.

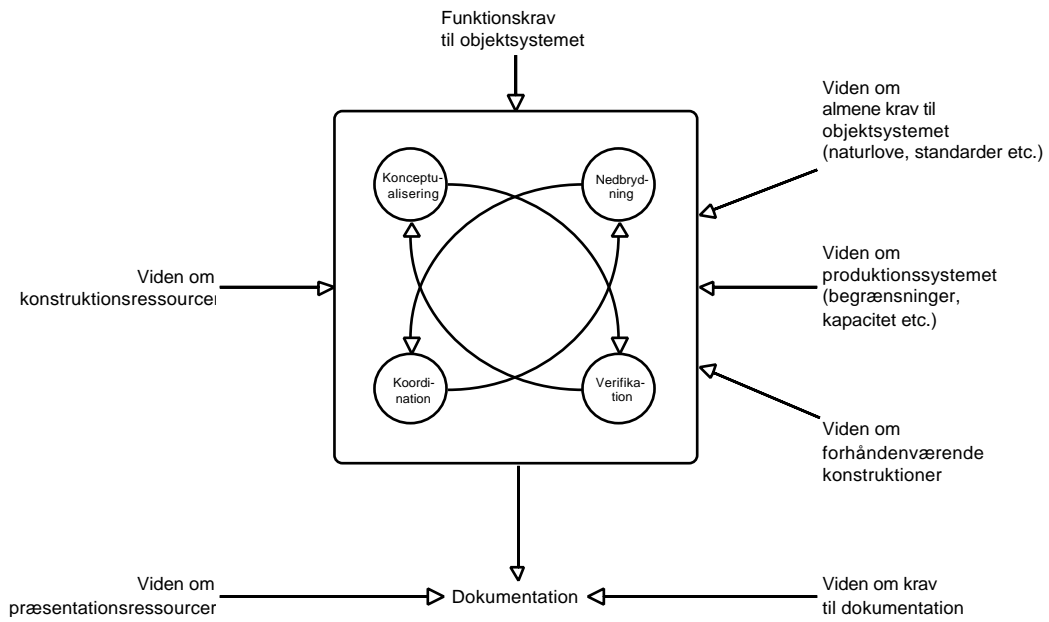


Fig. #. Model af teknikerens arbejde. På højre fløj er anført teknikerens viden om de tre problemområder: objektsystemet, produktionssystemet og de forhåndenværende konstruktioner. På venstre fløj er anført teknikerens viden om sine ressourcer som tekniker.

3. Teknikerens Integrerede Arbejdsplads

På grundlag af den ovenfor anførte analyse af de almene karakteristika ved teknikerens arbejde kan vi nu specificere nogle af de grundlæggende krav til Teknikerens Integrerede Arbejdsplads.

En arbejdsplads er ikke helt det samme som en arbejdsstation. Begrebet om en arbejdsplads er mere omfattende end begrebet om en arbejdsstation. Teknikerens arbejdsplads er det system af ressourcer (maskinel, programmel, arkiver, faciliteter i øvrigt), der støtter ham i hans arbejde. Teknikerens Integrerede Arbejdsplads kan således defineres som *et datamat-integreret støttesystem for teknikere*. (Begrebet Teknikerens Integrerede Arbejdsplads svarer til det af Donald Knight foreslåede begreb: Engineering Support System).²³

TIA bør ikke opfattes som noget isoleret, som et koncept for en "enkeltbruger" arbejdsstation, men som et led i den igangværende bevægelse i retning af datamat-integrerede produktionssystemer (CIM). Teknikerens arbejdsplads er i færd med at blive integreret med produktionsapparatet og det administrative system ved hjælp af et datamatbaseret system. TIA vil derfor skulle kunne give adgang til at importere data, der genereres i produktionsprocessen, ligesom det skal være muligt at overføre f.eks. styklister og bearbejdningsanvisninger til produktionen: "for applications in the 'factory of the future' engineering support systems will have to be interfaced to physical processes in real time".²⁴

TIA vil samtidig skulle kunne give adgang til at im- og eksportere data fra og til de administrative funktioner.

3.1. Organisationens integrerede database

Teknikeren arbejder som anført i forhold til tre problemdomæner, idet teknisk arbejde alment indebærer

- viden om det endelige produkts funktioner og egenskaber (den ønskede tilstand);
- viden om det eksisterende produktionssystemes muligheder og begrænsninger;
- viden om eksisterende, potentielt anvendelige komponenter.

Teknisk arbejde består grundlæggende i at relatere og kombinere viden om disse tre problemdomæner.

TIA må bygge derfor på en eksplicit repræsentation af problemrummet. TIA må altså baseres på et databasesystem, hvori information vedrørende de tre problemdomæner er integreret. Da teknisk arbejde ydermere er kooperativt, må det basale integrerede databasesystem (principielt) omfatte alle organisationens arkivsystemer. Dette er der generel enighed om i litteraturen:

"A controlled database is the most essential element in the proper implementation of CAE."²⁵

²³ Donald O. Knight: "The Engineering Workstation and the Engineering Support System - Present Status, Future Directions", *IEEE Micro*, oktober 1985.

²⁴ Donald O. Knight: "The Engineering Workstation and the Engineering Support System - Present Status, Future Directions", *IEEE Micro*, oktober 1985.

²⁵ Les Holland m.fl.: "Engineering Support System Software", *IEEE Micro*, oktober 1985. - CAE, dvs. Computer Assisted Engineering.

“The software needs of users depend on their responsibilities and level within the organization. One basic requirement exists - information for all users should come from a common database that can be concurrently accessed and updated by all [...]. The database manager is considered by many to be the ‘hub’ of an ESS.”²⁶

TIA må betragtes som en ‘front end’ til Computer Integrated Manufacturing, eller med andre ord: TIA er en grænseflade mellem CIM-systemet og problemløsningsaktiviteter i forbindelse med teknisk planlægningsarbejde.

Etablering af datamatintegrerede støttesystemer til en teknisk organisation som f.eks. en fremstillingsvirksomhed er imidlertid en uhyre krævende opgave. I organisationens mangfoldige aktiviteter anvendes et uhyre omfattende og som regel meget flydende begrebsapparat. Betænk blot de store terminologiske forskelligheder mellem de forskellige involverede fagdiscipliner. Hvis de enkelte teknikere skal kunne trække på data, der er blevet skabt af andre, er det således nødvendigt, at arbejdsdomænet analyseres, og at der udvikles en fælles begrebsverden for de involverede aktiviteter. Det forudsætter med andre ord, at der udvikles et konsistent og dækkende begrebsapparat for organisationen.

De data, organisationens integrerede database indeholder, skal bruges i mange, fundamentalt forskellige opgavesammenhænge: konstruktion, procesplanlægning, produktionsstyring, administration etc. og med forskellige formål for øje: planlægning, styring, evaluering etc. Dette betyder, at data skal kunne præsenteres på en for de forskellige opgaver indholdsmæssigt relevant måde og på et for den givne beslutningsfunktion relevant abstraktionsniveau. Det er således, som påpeget af Jens Ramussen

“nødvendigt, at information om det faktiske sagsindhold i arbejdsdomænet er tilgængeligt på flere beskrivelsesniveauer, og man skal kunne søge ud fra flere synsvinkler (set som mål, middel, eller direkte som funktion. [...]) Repræsentation af viden i systemet må tage hensyn til, at beslutninger tages på forskellige niveauer, og at den nødvendige information skal kunne findes fra forskellige synsvinkler med spørgsmål som hvorfor, hvad, hvordan? Database indhold skal derfor tilpasses mål-middel relationer, der er karakteristiske for arbejdsdomænet, og kan ikke skabes ved at samle forskellige eksisterende databaser og tilpasse dem til brugerne med en ‘interface’.”²⁷

Den kooperative karakter af teknikerens arbejde indebærer derudover behov for faciliteter til udveksling af meddelelser og dokumenter, deling af dokumenter osv.

3.2. Teknikerens arbejdsstation

Ved en arbejdsstation forstås en datamat, der står til rådighed for et individ, og som råder over så stærke faciliteter, at den overvejende del af brugerens arbejds-opgaver kan udføres ved hjælp af arbejdsstationen, frem for alt de i den givne sammenhæng centrale opgaver.

“Avancerede informationsteknologiske arbejdsstationer skal”, som Jens Rasmussen har udtrykt det, “kunne støtte brugere i overordnede beslutningssituationer, hvor problemløsning og improvisation er nødvendig, og for hvilke hensigtsmæssige procedurer ikke kan opstilles på forhånd.”²⁸

²⁶ L. W. Bezanson m.fl.: “Engineering Support System User Requirements”, *IEEE Micro*, oktober 1985.

²⁷ Jens Rasmussen: “Menneske-maskin systemer. Principper for design af intelligente systemer”, Indlæg på FTU-Seminaret, 27.-28. august 1986, DTH, Lundtofte, p. 3.

²⁸ Jvf. Jens Rasmussen: “Menneske-maskin systemer. Principper for design af intelligente systemer”, Indlæg på FTU-Seminaret, 27-28 august 1986, DTH Lundtofte.

En arbejdsstation er ikke en maskine, man betjener, men en maskine, man anvender.

Normalt forbinder man med begrebet arbejdsstation, at der er tale om en generel eller multifunktionel datamat, til forskel fra en "dum" terminal forbundet med en værtsdatamat. Da prisen på regnekraft og datalager i disse år falder meget hastigt og sandsynligvis vil vedblive at falde år frem i tiden, er det ved planlægningen af udviklingen af teknikerens arbejdsplads fuldt ud realistisk at gå ud fra, at det - i hvert fald når planen føres ud i livet - er økonomisk rentabelt, at alle eller praktisk talt alle opgaver i forbindelse med teknisk problemløsning udføres lokalt, på den enkelte teknikers arbejdsstation. Prisen på datatransmission er godt nok også faldende, men da den ikke falder med samme hast som prisen på regnekraft og datalager, kan man roligt antage, at distribution af regnekraft og datalager bliver økonomisk mere og mere fordelagtigt. En konfiguration baseret på en værtsdatamat med en stribe terminaler eller en konfiguration, hvor en værtsdatamat får overdraget de tunge opgaver fra mindre personlige datamater, som disse ikke kan magte på acceptabel tid, er dermed i færd med at miste økonomisk rationalitet.

3.2.1. ITERATIONSSTØTTE

Vi har set, at teknisk arbejde har en udpræget iterativ karakter. Dette gælder teknisk arbejde overhovedet (dvs. tekniske systemers livscyklus) såvel som den enkelte teknikers problemløsningsopgave. Den enkelte teknikers arbejde har i bund og grund karakter af en iterativ proces.

Teknikerens arbejdsstation skal understøtte processens iterative karakter.

Den skal først og fremmest i sin funktionsmåde støtte en *interaktiv arbejdsmåde*, dvs. at den dialogstil, der karakteriserer interaktionen mellem brugeren og arbejdsstationen, skal være interaktiv (direkte manipulation etc.). At arbejdsstationen skal understøtte en interaktiv arbejdsmåde stiller ikke alene krav om en interaktiv dialogstil, men stiller tillige krav om, at teknikeren kan anvende arbejdsstationen til at søge efter løsninger, evaluere evt. egnede løsninger, verificere løsningerne ved simulation eller afprøvning, nedbryde problemet i delproblemer, koordinere deløsningerne etc., med andre ord, at arbejdsstationen giver adgang til applikationsprogrammer af forskellig art, der kan støtte selve beslutningsprocessen. Det vil f.eks. sige simulationsprogrammer, videnbaserede systemer samt i øvrigt applikationer, der ved at anskueliggøre teknikerens tanker for ham (tegning, tekstbehandling) ligeledes bidrager til beslutningsprocessen.

TIA vil endvidere støtte det tekniske arbejdes iterative karakter ved at gøre iterationerne mindre arbejdskrævende såvel som mindre tidskrævende. Det bliver dermed på den ene side muligt at øge antallet af mulige iterationer, øge antallet af alternative løsninger under overvejelse og dermed øge kvaliteten af produkter eller på den anden side at reducere gennemløbstiden og omkostningerne.

Nærmere bestemt skal TIA tilvejebringe faciliteter, der direkte understøtter iterationsprocessen ved at give teknikeren mulighed for interaktivt og meget hurtigt at koncipere en løsning og evaluere dens konsekvenser, f.eks. i form af simulationsprogrammer. Det bør i denne forbindelse bemærkes, at programmer af denne kategori stiller store krav til hardware. Svartiderne skal være meget korte (minutter, ikke timer), hvis simulationsprogrammer o.l. skal kunne støtte beslutningsprocessen. Men ved konstruktion af komplekse systemer vil simulationer være så langsomme, at de kun i begrænset omfang vil kunne understøtte en interaktiv arbejdsmåde. Arbejdsstationer, der skal understøtte en

interaktiv arbejdsmåde ved konstruktion af komplekse tekniske systemer, vil derfor f.eks. skulle udstyres med specialiserede datamaskiner, der støtter arbejdsstationens basale generelle datamaskine ved varetagelsen af disse særligt krævende opgaver, f.eks. simulationsacceleratorer.²⁹

TIA vil desuden reducere forbruget af arbejdskraft og tid til informationsbehandling per iteration, bl.a. ved at gøre det til en opgave af forsvindende omfang at foretage rettelser som følge af ændringer o.l. i form af udarbejdelse af nye konstruktionstegninger, gennemføre nye beregninger, foretage ændringer af rapporter, udarbejde nye styklister, produktionsplaner osv. Med de samme ressourcer kan man derved på samme tid gennemføre et større antal iterationer.

Endelig vil TIA øge den informationsbehandlingskapacitet, der står til rådighed for den enkelte beslutningstager såvel som for kollektivet, ligesom TIA - i og med arbejdsstationens funktionalitet - øger den enkelte teknikers faglige spændvidde. I begge tilfælde vil TIA (alt andet lige) bevirke en mindskelse af opsplitningen af arbejdet på individer og dermed en mindskelse af behovet for kommunikation mellem disse ved løsningen af en given opgave. TIA vil også dermed indebære, at antallet af iterationer, der kan gennemføres med givne ressourcer inden for en given tidsramme, kan øges.

3.2.2. FUNKTIONALITETSSPEKTRUM

Indholdet i teknikerens arbejde er problemløsning i forbindelse med planlægning af tekniske systemer.

Som anført er den enkelte teknikers arbejde meget sammensat i den forstand, at han i sit arbejde udfører opgaver af kvalitativt forskellig karakter, f.eks. tegning, beregninger, simulation, forberedelse af forsøgsopstilling, afvikling af forsøg, analyse af forsøgsdata, udfærdigelse af rapporter og teknisk dokumentation, korrespondance, projektplanlægning, projektreghnskab osv. TIA må derfor frembyde faciliteter, der støtter hele dette brede spektrum af opgaver. Eftersom datamatbaserede beslutningsstøttesystemer vil skabe betingelser for at udvide den enkelte teknikers arbejdsfelt, vil det spektrum af opgaver, TIA skal understøtte, endda blive bredere, end man umiddelbart skulle formode på grundlag af den i dag foreliggende situation.

Antallet af applikationsprogrammer af relevans for den enkelte teknikers arbejde vil således være stort. TIA må derfor give adgang til et meget stort udbud af applikationsprogrammel.

1) Først og fremmest programmel til understøttelse af de centrale opgaver i forbindelse med teknisk problemløsning.

Et beslutningsstøttesystem til teknisk arbejde må have adgang til og kunne behandle viden vedr. de tre problemområder, der udgør det tekniske arbejdes problemrum:

- viden om kravene til objektsystemets egenskaber, dvs. viden om de funktionsbetingelser, som opgaveomverdenen indebærer, viden om de fysiske etc. lovmæssigheder objektsystemets "indre omverden" er underlagt, viden om standarder;
- viden om fremstillingssystemets kapacitet, tolerancer etc.;

²⁹ Jvf. Paul Wallich: "A review of engineering workstations", *IEEE Spectrum*, vol. 21, nr. 10, oktober 1984, pp. 48-53, og Paul Wallich: "Engineering workstations: a field report", *IEEE Spectrum*, vol. 21, nr. 12, pp. 52-56.

- viden om eksisterende konstruktioner, modeller etc.

Et beslutningsstøttesystem skal typisk have en grafisk brugergrænseflade, således at brugeren kan skitsere den konceptuelle model af objektsystemet i grafisk form, evt. ved valg blandt et udvalg af symboler. Under alle omstændigheder må interaktionen mellem brugeren og et beslutningsstøttesystem foregå interaktivt.

Brugeren vil typisk meddele beslutningsstøttesystemet de særlige betingelser, hvorunder objektsystemet skal fungere. På grundlag heraf vil systemet f.eks. på grundlag af de modeller, standardparametre etc., som det bygger på, kunne foreslå en løsning. Hvis den foreslåede løsning ikke er tilfredsstillende, ændrer brugeren løsningen på de punkter, hvor den er tilfredsstillende, og systemet anmoder så brugeren vurdere og evt. ændre de parametre, der er i strid med den af brugeren justerede løsning. Systemet er i stand til at udfærdige tegninger af det foreslåede system på flere abstraktionsniveauer.³⁰

Et beslutningsstøttesystem skal ofte skulle omfatte værktøj til simulation af det konceptuelle systems adfærd under de forventelige funktionsbetingelser såvel som til verifikation af komplekse systemers indre konsistens (f.eks. logisk simulation ved konstruktion af integrerede kredsløb). Omfatter beslutningsstøttesystemet sådanne simulationsværktøjer, vil iterationerne søgning-afprøvning og nedbrydning-koordination kunne understøttes fuldt ud.

2) Teknisk problemløsning omfatter tillige en mængde hjælpeopgaver af teknisk art, hvortil svarer et stort antal applikationer:

- Beregning som f.eks. numerisk analyse.
- Teknisk tegning til dokumentationsformål o.l.
- Datafangst i forbindelse med afprøvning af prototyper o.l.
- Statistisk og grafisk behandling af testdata.
- Regneark.

Af disse er nogle applikationer opgavespecifikke, mens andre (e.g. regneark) er generelle.

3) Eftersom det tekniske arbejde udviser meget stor spredning i henseende til opgavernes art, vil tekniske organisationer typisk komme ud for at skulle udvikle applikationsprogrammer til eget brug. Som oftest mindre beregningsprogrammer eller modeller kodet i FORTRAN. Man må endvidere antage, at tekniske beslutningsstøttesystemer som regel vil skulle udvikles af organisationen selv. Arbejdsstationen skal derfor give adgang til et eller flere programmeringssprog med tilhørende udviklingsværktøj (prototypeværktøj, editor, oversætter, debugger).

4) Ved siden af disse de centrale tekniske problemløsningsopgaver og de dertil hørende hjælpeopgaver udfører teknikere et væld af opgaver af administrativ art, som påkalder sig applikationer af samme type som dem, man anvender i konventionelle administrative sammenhænge:

- Tekstbehandling.
- Elektronisk post.
- Arkivering og genfindning af dokumenter.
- Informationssøgning.
- Projektplanlægning.

³⁰ Et system af denne art er implementeret i tilbudsafdelingen hos F. L. Smidth & Co., hvor det anvendes til konfiguration af store subsystemer. Systemet har betydet, at et tilbud, det tidligere tog uger at udarbejde, nu kan udarbejdes på en halv dag; at det er muligt at udarbejde flere alternative forslag, samt at den enkelte teknikers arbejdsfelt er blevet udvidet. - Det pågældende system kunne producere et prisoverslag, der dækkede ca. 90% af samtlige omkostninger.

Udbuddet af applikationsprogrammer til teknikerens arbejdsstation vil således normalt skulle være langt større, end det er tilfældet ved kontorsystemer.

3.2.3. INTEGRATION AF OPGAVER

Eftersom teknikerens arbejde er karakteriseret ved et bredt opgavespektrum, og eftersom antallet af applikationer, som arbejdsstationen skal give adgang til, er meget stort, skal teknikerens arbejdsstation skal for det første give adgang til nemme og smertefrie skift mellem et relativt stort antal programmer. Dette vil reelt kræve multitasking. Dette lægger på sin side op til et operativsystem baseret på Unix eller lignende, hvilket igen kræver en masse intern hukommelse (> 2-4 Mb RAM).

Af samme grund skal teknikerens arbejdsstation for det andet muliggøre nem og smertefri im- og eksport af data mellem applikationsprogrammer. Da det brede spektrum af opgaver tillige implicerer arbejde med et meget stort antal forskellige datatyper, skal det være muligt efter behov at integrere forskellige datatyper i ét og samme dokument. Operativsystemet skal være forberedt herpå i form af et fælles arbejdslager med standarder for formater af data, der skal udveksles herigennem (f.eks. i stil med Clipboard på Apple's Macintosh). Der påhviler i det hele taget leverandøren af arbejdsstationen et stort ansvar for at definere konventioner for dataformater og vejlede de programmelhuse, der udvikler applikationsprogrammel til arbejdsstationen, i brugen af disse konventioner. (Jvf. Apple's *Inside the Macintosh*, vol. 1-5).

Programmel, der alligevel ikke overholder sådanne konventioner, kan evt. integreres i TIA ved hjælp af specielt programmel, der reformatterer filer. Behovet for reformattering vil nødvendigvis opstå, når man vil udveksle filer mellem arbejdsstationer af forskelligt fabrikat (så længe de forskellige leverandører ikke overholder almene standarder). Sådanne reformatteringsprogrammer er ved at blive tilgængelige kommercielt, som "plastikprogrammel", men der vil uvægerligt opstå behov for reformattering, der ikke kan tilfredsstilles med plastikprogrammel. Dette problem kan afhjælpes med ad hoc programmel, udarbejdet med et højniveausprog, f.eks. et macrosprog. Denne løsning kræver dog, at producenter af applikationsprogrammel offentliggør specifikationerne af de filformater, der anvendes de de forskellige applikationsprogrammer.

Teknikerens arbejde er ikke alene uhyre sammensat, men tillige varierende i den forstand, at opgavernes karakter varierer fra den ene tekniker til den anden, og at opgavernes karakter varierer over tid. Teknikerens arbejdsstation skal derfor være åben over for integration af nyt programmel. Dette krav understreger de ovenfor anførte krav til integration - og kan imødekommes på samme vis.

3.2.4. BRUGERGRÆNSEFLADE

Arbejdsstationens brugergrænseflade omfatter dels operativsystemets brugergrænseflade og dels de enkelte applikationers brugergrænseflade.

Udformningen af arbejdsstationens brugergrænseflade er af kritisk betydning for, om de muligheder, TIA rummer, vil blive realiseret.³¹

³¹ "Amid [the] barrage of new technology there remains one fundamental aspect that demands more attention that it has received until now, namely the Man Machine Interface (MMI), how does an engineer best interact with the computer? If this problem is not tackled, as hardware and software becomes cheaper, we will arrive at the point where the major cost of using computer systems for

Sagen er i virkeligheden ofte den, at en teknisk organisation implementerer et datamatbaseret system til understøttelse af teknisk arbejde. Systemet er teknisk set af høj kvalitet (stabilit, integreret etc.), men ikke desto mindre bliver det ikke anvendt i det forudsete eller mulige omfang. Efter at have implementeret systemet skal man altså til at missionere blandt de vantro og omvende dem til brugere af systemet.

Problemet skyldes ikke, at de skeptiske teknikere handler irrationelt. De handler faktisk rationelt ud fra deres situation. De anlægger en kølig cost-benefit analyse og vurderer, at systemet kræver mindst lige så meget arbejde af dem, som de kan forvente at spare ved at bruge det.

Teknikeren er sjældent edb-ekspert. Brugergænsefladen må derfor ikke forudsætte, at han eller hun er det eller har tid og lyst til at blive det. Systemets dialog med brugeren må være således udformet, at den er intuitivt forståelig for "lægmand".

Under alle omstændigheder består teknikerens arbejde i andre opgaver end lige netop at anvende en datamat. Datamaten er med andre ord blot et redskab. System skal derfor være "transparent" i forhold til opgavens indhold. Dvs. dets "indre omverden" bør så vidt muligt ikke kræve mentale operationer af brugeren, der er fremmede for opgavens indhold.

Teknikerens arbejde består i teknisk problemløsning, ikke problemløsning i forhold til det edb-udstyr, han anvender. Arbejdsstationens programmel skal understøtte teknikeren i selve arbejde, ikke distrahere fra det.

Eftersom teknikerens arbejde udpræget har iterativ karakter, må arbejdsstationens brugergænseflade som allerede anført understøtte en interaktiv dialogstil, således at teknikeren umiddelbart og anskueligt får tilbagemelding om konsekvenserne af påtænkte løsninger.³²

Teknikeren har behov for adgang til et væld af applikationsprogrammer. Interaktionen mellem brugeren og de forskellige applikationsprogrammer bør derfor - i den udstrækning, hvori applikationsprogrammerne repræsenterer samme eller beslægtet funktionalitet - være ensartet. De forskellige applikationers brugergænseflade må med andre ord fremtræde ensartet i den forstand, at den samme kommando giver samme resultat i de forskellige programmer.

Teknikeren anvender ikke alene mange applikationsprogrammer, men anvender dem med uregelmæssige mellemrum. De operationer, teknikeren skal udføre for at kunne få adgang til applikationernes funktioner, skal derfor i sig selv være lette at erindre eller genkalde. De i den givne sammenhæng relevante funktioner skal præsenteres for brugeren i en let genkendelig form (menu, ikon).

Moderne avancerede arbejdsstationer indfrier i høj grad disse krav. De fire grundlæggende principper i en sådan brugergænseflade blev fastlagt i forbindelse med udviklingen af Xerox Star arbejdsstationen: 1) For at give datamaten en kommando behøver brugeren ikke "huske og skrive", men kan nøjes med at "se og

engineering applications will become the cost of the user. As we approach this stage the financial gains will be achieved not from striving for more effective use of the computing resource but from effective use of the user which is closely linked to the MMI." (P.F. Arthur: "The Engineering workstation: some thought on its future form", *Computer-Aided Design*, vol. 17, nr. 3, april 1985, p. 115).

³² Jvf. Ben Shneiderman: "Direct Manipulation: A Step Beyond Programming Languages", *Computer*, vol. 16, nr. 8, august 1983, pp. 57-69, og Edwin L. Hutchins, James D. Hollan and Donald A. Norman: "Direct Manipulation Interfaces", i D. A. Norman and S. W. Draper (ed.): *User Centered System Design. New Perspectives on Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, New Jersey, -London, 1986, pp. 87-124.

pege". Funktioner og dokumenter præsenteres grafisk for brugeren, der med en "mus" kan udpege det valgte. 2) Datamatens kompleksitet reduceres i forhold til brugeren ved at denne udelukkende præsenteres for valgmuligheder, der er relevante i den givne situation. 3) Kommandoerne er ensartede på forskellige områder. Har man først forstået systemets logik, kan man således sjusse sig til, hvad man skal gøre, uden at gribe til manualen. 4) Det man ser på skærmen, svarer til det trykte dokument. ("What You See is What You Get" eller WYSIWYG).

Disse principper er blevet lagt til grund for udformningen af brugergrænsefladen på moderne arbejdsstationer som f.eks. Xerox 6085 og Apple's Macintosh. For Macintosh'ens vedkommende er det tilmed lykkedes at sikre, at de forskellige programmehuse, der i indbyrdes konkurrence producerer applikationsprogrammer til denne datamat, overholder et sæt konventioner for udformningen af applikationsprogrammernes brugergrænseflade. Brugeren arbejder derfor i en relativt ensartet verden, hvilket gør det let at lære at anvende et nyt program.

Principperne har dannet skole og forsøges nu - i den udstrækning, det er muligt - overført til IBM PC'er, jvf. f.eks. fremkomsten af programpakker som *GEM* og Microsoft *Windows*.

Det afgørende kriterium for en brugergrænseflades kvalitet er givetvis, at systemet frembyder mindst mulig "modstand" for tankevirksomhed af værdi for den opgave, brugeren skal løse. Dvs. at systemet ikke distraherer, sinker eller forstyrrer tankevirksomheden. Når man f.eks. skriver i hånden, kræver selve betjeningen af værktøjet ikke megen omtanke. Rettelser kan foretages meget hurtigt. Dette gør blyanten til en klassiker. Skriver man i hånden, er skriveprocessen imidlertid meget langsommere end tankeprocessen. Tanken iler forud for nedfældelsen på papir, og man kommer uvægerligt i den kedelige situation, at den epokegørende idé, man kom på under nedfældelsen af en allerede formuleret sætning, er glemt, når man når frem til at skulle formulere den nye idé. Skrivemaskinen derimod, er et langt hurtigere skriveredskab. Problemet er her, at redskabet tvinger brugeren ind i en strengt sekventiel skrivemåde, der grundlæggende er uforenelig med kreativ tænkning. Forfattere, der benytter en skrivemaskine, flår derfor i ét væk papiret ud af maskinen, krøller det sammen og kyler det i papirkurven - og tyr så i øvrigt til saks og klister.

En arbejdsstation har en god brugergrænseflade, hvis brugeren ikke bemærker den. Teknikerens arbejdsstation skal så vidt muligt forskåne brugeren for enhver dialog med systemet, som ikke har direkte relation til indholdet af sagsbehandlingsarbejdet. Alt andet er sagen uvedkommende og kan kun reducere den mulige gavnlige effekt af datamatstøttet sagsbehandling.

Et eksempel. Som allerede anført består omkostningerne ved et datamatbaseret system ikke alene af udgifterne til maskinel og programmel. Der er mange andre poster, der skal tages i betragtning. Ikke mindst uddannelsesomkostningerne. Tager man systemer med konventionel kommando- og menu-baseret menneske-maskine dialog (typisk maskiner med MS-DOS styresystem, f.eks. IBM's PC) og sammenligner med systemer med avanceret menneske-maskine dialog (som f.eks. Apple's Macintosh og Xerox arbejdsstationerne), viser forskellen i uddannelsesomkostningerne sig at være betragtelig.³³ Antager man, at omkostningerne for en medarbejder under uddannelse beløber sig til 1.500 kr. pr dag, ser regnestykket nogenlunde således ud:

³³ Åke Engquist, Sigurd Eriksson, Roger Lindström, Bernt Palmqvist: *Boken om kontorsautomation*, Studentlitteratur, Lund, 1985, pp. 101-102.

	Konventionelle systemer		Avancerede systemer	
	Dage	Kr.	Dage	Kr.
Uddannelse i grundfunktionerne	15	22.500	5	7.500
Uddannelse i hver yderligere funktion	5	7.500	1	1.500

Forskellen i uddannelsesomkostningerne mellem et konventionelt system og et system med en avanceret brugergrænseflade er altså for grundfunktionerne 10 dage eller 15.000 kr. Dertil kommer en difference på fire dage for hver applikation. Hvis antallet af funktioner ud over grundfunktionerne f.eks. blot er to, ser forskellen i uddannelsesomkostningerne således ud:

	Konventionelle systemer		Avancerede systemer	
	Dage	Kr.	Dage	Kr.
Uddannelse i yderligere funktioner, i alt	10	15.000	2	3.000
Uddannelsesomkostninger, i alt	25	37.500	7	10.500

Den samlede forskel i uddannelsesomkostninger på de to typer af systemer beløber sig i så fald til 18 dage eller 27.000 kr. Hvis systemet anvendes af to medarbejdere, bliver forskellen naturligvis den dobbelte, nemlig 36 dage eller 54.000 kr. Endelig er det realistisk at antage, at der i systemets levetid vil ske en udskiftning af medarbejderne. Antager vi, at de to medarbejdere, der opridelig oplæres i brugen af systemet, på et tidspunkt i systemets levetid erstattes af to andre, bliver den samlede besparelse altså på 108.000 kr.

Den umiddelbare merudgift til systemer med avancerede brugergrænseflader kan således bestemt forsvares, også økonomisk.

Hertil kunne man indvende, at selv om et system er nemt og hurtigt at lære, er det tænkeligt, at det alligevel ikke er effektivt i daglig brug. Det er måske distraherende og langsommeligt, og måske indbyder det til fejl. Indvendingen virker umiddelbart tilforladelig. Den synes imidlertid ikke at være holdbar. Grundige undersøgelser af ni forskellige tekstbehandlingsprogrammer viser på, at der ikke er nogen modsætning mellem, at systemet er nemt at lære for nybegyndere, og at det er et effektivt redskab for den øvede bruger. Der var faktisk en positiv sammenhæng mellem, hvor nemt det er at lære, og hvor nemt det er at anvende.³⁴ Grunden til dette er nok først og fremmest, at et program er nemt at lære, hvis dets opbygning virker logisk og gennemskuelig. I så fald vil også den øvede bruger kunne arbejde mere koncentreret med systemet og være mindre tilbøjelig til at begå fejl.

Dertil kommer, at teknikeren jo vil som regel ikke anvende det samme applikationsprogram dag ud og dag ind. Der vil ofte kunne gå flere dage mellem f.eks. skriveopgaver, ligesom teknikeren vil skifte mellem mange applikationsprogrammer. Jo mere logisk og gennemskuelig systemets opbygning virker, i desto højere grad vil det derfor være effektivt og sikkert i daglig drift.

³⁴ Ronald Baecker og William Buxton: *Readings in Human-Computer Interaction: A Multidisciplinary Approach*, Morgan Kaufmann Publishers, 1987, p. 217.

Endelig vil en dårlig brugergrænseflade medføre, at teknikeren vil have modvilje mod systemet og være tilbøjelig til at undgå at anvende det. Hvorfor anvende et bestemt redskab, hvis man med et andet redskab kan udføre opgaven lettere eller hurtigere? Så selv om systemet måske har været relativt billigt, kan det jo i så fald dårligt forrente sig.

Den økonomiske fordel ved et system med en avanceret grafisk brugergrænseflade er derfor ikke mindre, men større end det fremgår af opstillingen ovenfor.

Brugergrænsefladen skal være fleksibel sådan at forstå, at den skal kunne tilpasses den enkelte brugers færdigheder og behov. Den uøvede bruger af et tekstbehandlingsprogram vil f.eks. typisk have behov for en menubaseret dialogstil. Senere, når brugeren behersker programmet, vil brugeren givetvis foretrække at kunne skyde genveje, f.eks. ved brug af funktionstaster, hvorved brugeren giver systemet en kommando ved en enkel tasteoperation. Den øvede bruger vil endvidere kunne have gavn af mulighed for på egen hånd at kunne skrive "macroer", dvs. forkortelser, der aktiverer en hel række funktioner.

For at reducere kompleksiteten af interaktionen og derved reducere den mentale belastning og fejlhyppigheden skal de i den givne sammenhæng irrelevante funktioner skjules for brugeren.

Eftersom teknisk arbejde er udpræget iterativ, gennemløbes den enkelte opgave på mange niveauer, i en fremadskridende spiralbevægelse. Teknikeren vil derfor ofte have behov for at anvende et applikationsprogram på flere niveauer. Brugeren skal derfor have mulighed for at "skjule" de mere avancerede funktioner i et applikationsprogram.

Et af disse punkter fortjener en mere indgående diskussion. Hvordan sikres den nødvendige ensartethed i interaktionen mellem brugeren og de forskellige applikationsprogrammer, han anvender? En oplagt idé er, at "indskyde" et særligt program mellem brugeren og applikationerne, der formidler interaktionen mellem de forskellige applikationer og brugeren. (Se fig. #).³⁵

Vil et sådant "mægler"program kunne sikre, at interaktionen mellem bruger og applikation opnår en rimelig grad af ensartethed?

³⁵ Nogle forfattere kaldt en sådant "mægler"program et User Interface Management System. Vi er enige med bl.a. William Buxton, når han reserverer denne betegnelse til et sæt af udviklingsværktøj, ved hjælp af hvilket programmøren støttes i udformningen og implementeringen af en applikations brugergrænseflade. Jvf. bl.a. Dan R. Olsen, William Buxton, Roger Ehrich, David J. Kasik, James R. Rhyne and John Sibert: "A Context for User Interface Management", *IEEE Computer Graphics and Applications*, vol. 4, nr. 12, december 1984, pp. 43-50.

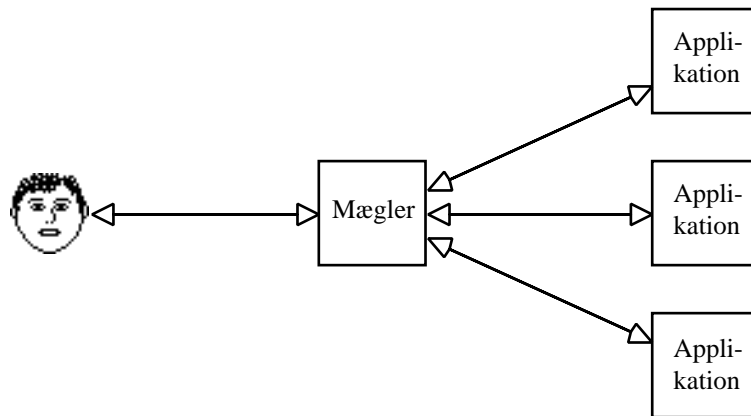


Fig. #. Et særligt “mægler”program formidler i denne arkitektur interaktionen mellem brugeren og de forskellige applikationsprogrammer.

Denne arkitektur er imidlertid problematisk. Idéen bygger på en overfladisk forståelse af arbejdssituationen. Brugers interaktion med maskinen er i grunden ikke en interaktion med maskinen, men med den opgave, der skal udføres. Det er opgaven, der er indholdet i arbejdssituationen. Maskinen er et indskudt led mellem arbejderen og opgaven.

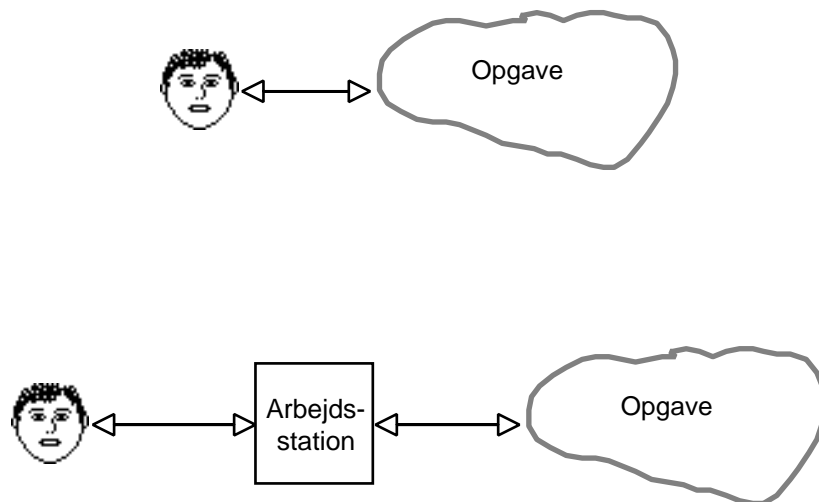


Fig. #. Arbejdsstationen er et indskudt led mellem arbejderen og opgaven. Interaktionen mellem arbejder og maskine (arbejdsstation) formidler interaktionen mellem arbejderen og opgaven.

Menneske-maskine interaktionen formidler interaktionen mellem opgaven og den, der udfører den. Applikationsprogrammet repræsenterer den for opgavens udførelse nødvendige funktionalitet og er med andre ord særdeles opgavespecifik., og de forskellige applikationsprogrammer repræsenterer opgaver af forskellig art. Interaktionens form er grundlæggende bestemt af opgavens indhold. Intet formidlingsprogram vil således kunne repræsentere alle tænkelige applikationers funktionalitet. Et formidlingsprogram “imellem” brugeren og applikationsprogrammet vil tværtimod være et filter mellem brugere og opgaven. Det vil standardisere interaktionen på bekostning af funktionalitet og transparens.

Et formidlingsprogram mellem operativsystemet og brugeren er derimod en nødvendighed. Operativsystemets funktionalitet er i virkeligheden brugerens

arbejde uvedkommende. Operativsystemet styrer datamatens “indre omverden”, som er aldeles uvedkommende for indholdet af brugerens arbejde. Mens brugeren har behov for “nærkontakt af tredje grad” i relation til applikationsprogrammernes repræsentation af opgavens indhold, har han behov for det stærkest mulige “filter” i relation til det univers, der er operativsystemets.

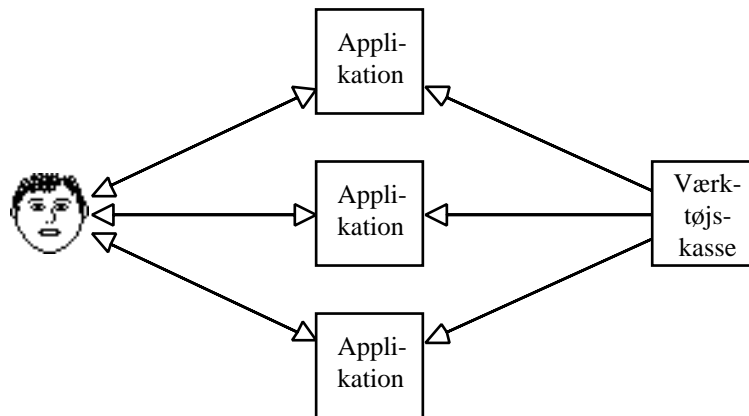


Fig. #. Brugeren er i nærkontakt med applikationern, der til gengæld trækker på en fælles “værktøjskasse”, der derved bidrager til at sikre en rimelig grad af ensartethed.

En alternativ arkitektur er illustreret i fig. #. Brugeren er her i nærkontakt med applikationsprogrammerne, der til gengæld trækker på en fælles “værktøjskasse” i form af moduler og præfabrikerede subrutiner (“ressourcer”) som f.eks. vinduer, rullegardinsmenuer og “pop up”-menuer. Denne værktøjskasse kan være en del af operativsystemet (evt. i hardware), som et sæt af ressourcer, applikationerne kan kalde på kørselstidspunktet, eller programmemoduler, der integreres i applikationerne i konstruktionsfasen. Det vil sige, at en programmør, der konstruerer et applikationprogram til en datamaskine, hvori en sådan værktøjskasse er indbygget, kan spare en mængde arbejde ved - i den udstrækning det ikke strider direkte med den krævede funktionalitet - at trække på de indbyggede ressourcer. (Man vurderer, at omk. halvdelen af kodelinjerne i et program medgår til styring af dialogen med brugeren, så der er meget at spare). Programmøren gøres med andre ord i vid udstrækning direkte personligt interesseret i at overholde de konventioner for udformingen af dialogen mellem bruger og maskine, som systemets designere har fastlagt.

Når de forskellige programmers brugergrænseflade ofte er meget forskellige, også på punkter hvor funktionaliteten ikke kræver det, skyldes det primært, at programmørerne ikke har nogen som helst interesse i standardisering af dialogen. Uden en værktøjskasse af ressourcer, vil standardisering af brugergrænsefladen blot kræve ekstra arbejde af programmøren, uden at give noget til gengæld.

Et User Interface Management System, dvs. et udviklingsmiljø, der understøtter de forskellige opgaver i udviklingen af brugergrænseflader til applikationsprogrammel, bl.a. ved at give programmøren nem adgang til at trække på de ressourcer, der er nedlagt i systemets værktøjskasse, vil øge programmørens gevinst ved at overholde konventionerne for menneske-maskine dialogen.³⁶ En teknisk organisation, hvis tekniske medarbejdere selv i nogen udstrækning selv

³⁶ Dan R. Olsen, William Buxton, Roger Ehrich, David J. Kasik, James R. Rhyne and John Sibert: “A Context for User Interface Management”, *IEEE Computer Graphics and Applications*, vol. 4, nr. 12, december 1984, pp. 43-50.

udvikler applikationer til eget brug, kan derfor med fordel ofre stor opmærksomhed på udvælgelsen af et egnet UIMS til understøttelse af den interne applikationsudvikling.

3.3. Understøttelse af kooperativt teknisk arbejde

Eftersom teknisk arbejde principielt udføres på grundlag af ufuldstændig viden, er det som anført nødvendigt at sikre, at problemet anskues ud fra flere synsvinkler, og at den enkeltes vurderinger konfronteres med evt. alternative vurderinger. Teknisk arbejde har med andre ord - som andet arbejde, der har karakter af problemløsning, - en fundamentalt kooperativ karakter.

Datamat-baserede faciliteter til understøttelse af kooperativt arbejde forefindes allerede:

- Elektronisk post.
- Integrerede databaser.
- Redskaber til understøttelse af kollektiv udarbejdelse af rapporter, f.eks. versionskontrol, grafisk markering af ændringer i teksten og indføjelser af kommentarer i særskilte vinduer knyttet til bestemte passager i teksten (f.eks. i stil med hypertext).
- Fremvisning af anvendelsen af en applikation, f.eks. en simulationskørsel, på andre arbejdsstationer som led i en diskussion.
- Elektronisk tavle, hvor man kan skrive og tegne i hånden, men samtidig har adgang til at arkivere tekst og tegning i digitalt format.

Datamat-understøttelse af kooperativt arbejde har indtil for nylig ikke været gjort til genstand for målrettet og systematisk forskning. Inden for de senere år er problemet dog blevet genstand for megen opmærksomhed og debat, men forskningen på dette felt befinder sig stadig på et indledende stadium.

Teknisk arbejde er som sagt kooperativt arbejde. Det er kollektiv problemløsning. Datamatstøttet teknisk arbejde betyder derfor på den ene side, at de enkelte teknikere skal kunne kommunikere indbyrdes ved hjælp af arbejdsstationerne. Til behandlingen af en opgave er der på den anden side behov for adgang til information fra mange kilder, fra samme afdelings arkiver, fra arkiver i andre afdelinger i organisationen, fra arkiver i andre organisationer og fra centrale arkiver.

Selv om praktisk talt alle den enkelte teknikers opgaver skal kunne udføres ved hjælp af den lokale arbejdsstation, kan arbejdsstationen altså ikke være isoleret. Et lokalt netværk gør det muligt for teknikerne at trække på et fælles arkiv og udveksle dokumenter via arbejdsstationerne.

Et sådant netværk skal være overordentligt fleksibelt. Teknisk arbejde er karakteriseret ved en bevægelig arbejdsorganisation. Der sker naturligvis vedblivende ændringer af personalets sammensætning i form af f.eks. nyansættelser, ligesom der vedblivende sker omplaceringer i form af f.eks. forfremmelser. I og med sådanne ændringer forandres ansvars- og opgavefordelingen. Men ansvars- og opgavefordelingen ændres tillige løbende, i takt med strømmen af opgaver. Hvem der arbejder sammen med hvem, ændres fra opgave til opgaver. Hvilke medarbejdere der har behov for - og adgang til - adgang til bestemte dokumenter, varierer uafledigt. Heller ikke hvad arbejdsorganisationen angår, er fremgangsmåden bestemt på forhånd. Et lokalt netværk skal afspejle denne bevægelige arbejdsorganisation og skal derfor kunne

ændres let og smertefrit. Netværkets styresystem skal kunne håndteres af teknikerne selv og af ledelsen, uden at involvere edb-afdelingen eller leverandøren. Dets brugergrænseflade skal derfor opfylde de samme krav som arbejdsstationens.³⁷

Kravet om fleksibilitet gælder imidlertid i mere omfattende forstand. De forskellige maskinelle systemer i organisationen skal kunne udveksle data, "lodret" (mellem f.eks. arbejdsstationer og værtsdatamaten) og "vandret" (mellem f.eks. forskellige arbejdsstationer). Hvordan sikrer man denne integration af arbejdsstationerne? Standardsvaret er her standardisering af organisationens maskinel. Man kan naturligvis sikre en integreret løsning ved simpelthen at *forbyde* indkøb af maskinel, der ikke umiddelbart er 100 pct. kompatibelt med det maskinel, man har i forvejen. Edb-afdelingen i virksomheder og institutioner dekretterer således ofte standarder, der alvorligt begrænser systemets fleksibilitet. Ofte dekretteres et bestemt operativsystem. I nogle tilfælde dekretteres endda maskinel fra en bestemt leverandør, der som regel er et stort og velkendt internationalt selskab. Opnår man ikke andet, så opnår man dog, at ingen kan beskyldes én for at have været vildt eksperimenterende.

Tillad mig at forsøge mig med en lignelse, der måske slet ikke er nogen lignelse. Den verden, vi lever i, er kompleks og foranderlig. Denne kompleksitet og foranderlighed skaber uvægerligt problemer for ethvert system, der skal fungere i denne verden. Et system kan principielt håndtere problemerne på to måder. Ved at gøre samkvemmet med omverdenen mindre komplekst og foranderligt eller ved intelligent tilpasning. En organisation kan på den ene side forsøge at standardisere og regulere samkvemmet med omverdenen og dermed gøre det mere enkelt og forudsigeligt. Bureaukratiske organisationer har i umindelige tider med stort held pålagt deres omverden bestemte regler for samkvemmet med den, ved f.eks. at forordne, at henvendelse til organisationen skal ske ad bestemte kanaler, ved hjælp af særlige formularer, inden for bestemte tidsfrister osv. Problemet ved denne måde at håndtere kompleksitet og foranderlighed på er naturligvis, at den er gennemført ufleksibel. Men verden er og bliver kompleks og foranderlig. Resultatet er uvægerligt modsætninger mellem organisationen og dens omverden; organisationen mister følingen med verden omkring sig, desorienteres, mister indflydelse. Organisationerne kan på den anden side forsøge at lære at orientere sig i omverdenen, registrere forandringerne hurtigt og tage højde for dem. Den fleksible strategi kræver med andre ord evne til problemløsning.

Et integreret informationssystem må ligesom en organisation være fleksibelt. Det må være åbent over for tilkobling af et stort udvalg af maskinel svarende til de varierende brugerkrav.

Ved at forordne et meget begrænset udvalg af maskinel og programmer tvinges teknikerne til at arbejde med redskaber, der ikke nødvendigvis svarer til opgavens behov og den enkeltes arbejdsstil. Prisen for den moderate og konservative løsning er således som regel, at slutbrugerne ikke får nær det udbytte af systemet, som de kunne have fået, og at de måske endda undlader at anvende det i det omfang, der var forudset. Og se, så er prisen jo høj!

Der har været gode og holdbare tekniske grunde til at udvise den største varsomhed med at tillade blandede systemer. Integrationen af maskinelle systemer kan imidlertid i dag opnås, uden at man må gribe til en sådan restriktiv standardiseringspolitik, nemlig ved hjælp af "porte" eller "broer", dvs. specialiserede

³⁷ Jonathan Seybold: "Apple's AppleShare file server: Do-It-Yourself Multi-User System", *The Seybold Report on Desktop Publishing*, vol. 1, nr. 6, 9. februar 1987, pp. 3-16.

datamater med tilhørende programmel, der automatisk oversætter de til nettet knyttede maskiners forskellige kommunikationsprotokoller til netværkets standard. Man kan derved frit udveksle filer (dokumenter, registre, grafik) mellem vidt forskellige maskiner. For de mest udbredte arbejdsstationers vedkommende er sådanne "porte" med tilhørende programmel på markedet i dag. For mange andre fabrikaters vedkommende er protokolprogrammel under fremskreden udvikling.

I dag er det f.eks. muligt for et ringe beløb og uden smerter at integrere IBM PC, XT, AT og lignende i AppleTalk lokalnettet (ved hjælp af produkter som *MacLink* og *Tops*). Dokumenter kan derved udveksles frit mellem PC'ere og Macintosh'er. Dokumenter på PC'ens pladelager kan vælges og åbnes fra en Macintosh, ved hjælp af dennes styresystem. Omvendt kan PC'en benytte en laserprinter, der er tilkoblet AppleTalk nettet, som printer.

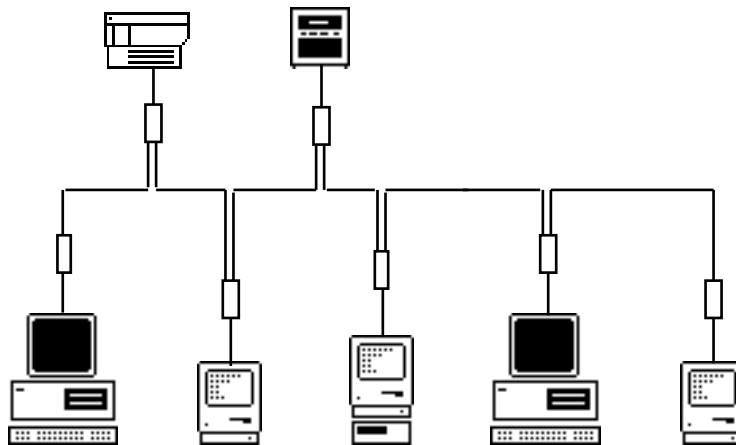


Fig. 10. Et lokalt netværk med arbejdsstationer af forskelligt fabrikat og med fælles faciliteter i form af en laserprinter og en hård disk. En konfiguration som den viste kostede i maj 1987 mindre end 250.000 kr., dvs. mindre end 50.000 pr. arbejdsstation.

Et andet eksempel, der i endnu højere grad dokumenterer den grad af fleksibilitet, der i dag er mulig: Ethernet lokalnettet, der blev udviklet af Xerox, er i dag en udbredt standard for kommunikation mellem datamater. De fleste datamater er i dag bygget til at kunne kommunikere via Ethernet eller kan udbygges med denne facilitet. Dette gælder datamater fra bl.a. Digital (DEC), Data General, Hewlett Packard, Siemens, Chr. Rovsing, Regnecentralen, DDE og naturligvis Xerox. I denne sammenhæng er det især interessant at bemærke, at avancerede arbejdsstationer som Xerox, Sun og Apollo's Domain datamater er forberedt for kommunikation via Ethernet. IBM PC'ere kan kobles direkte til sådanne arbejdsstationer. Det er tillige muligt at bygge bro mellem Ethernet og andre lokalnet. F.eks. er Kinetics' *FastPath* en programmerbar "port" mellem AppleTalk og Ethernet. Dette gør det muligt at lave en integreret løsning baseret på Ethernet med direkte tilkobling af Unix arbejdsstationer som Xerox og Sun og samtidig med tilkobling af et eller flere AppleTalk lokalnet med Macintosh og IBM PC mikro-datamater. *FastPath* gør det tillige muligt at forbinde AppleTalk nettet direkte med arbejdsstationer som f.eks. Sun og Apollo datamaterne, og ved hjælp af *FastPath* og *AlisaTalk* kan AppleTalk køre sammen med Digital's VAX/VMS minidatamat. Og så fremdeles.

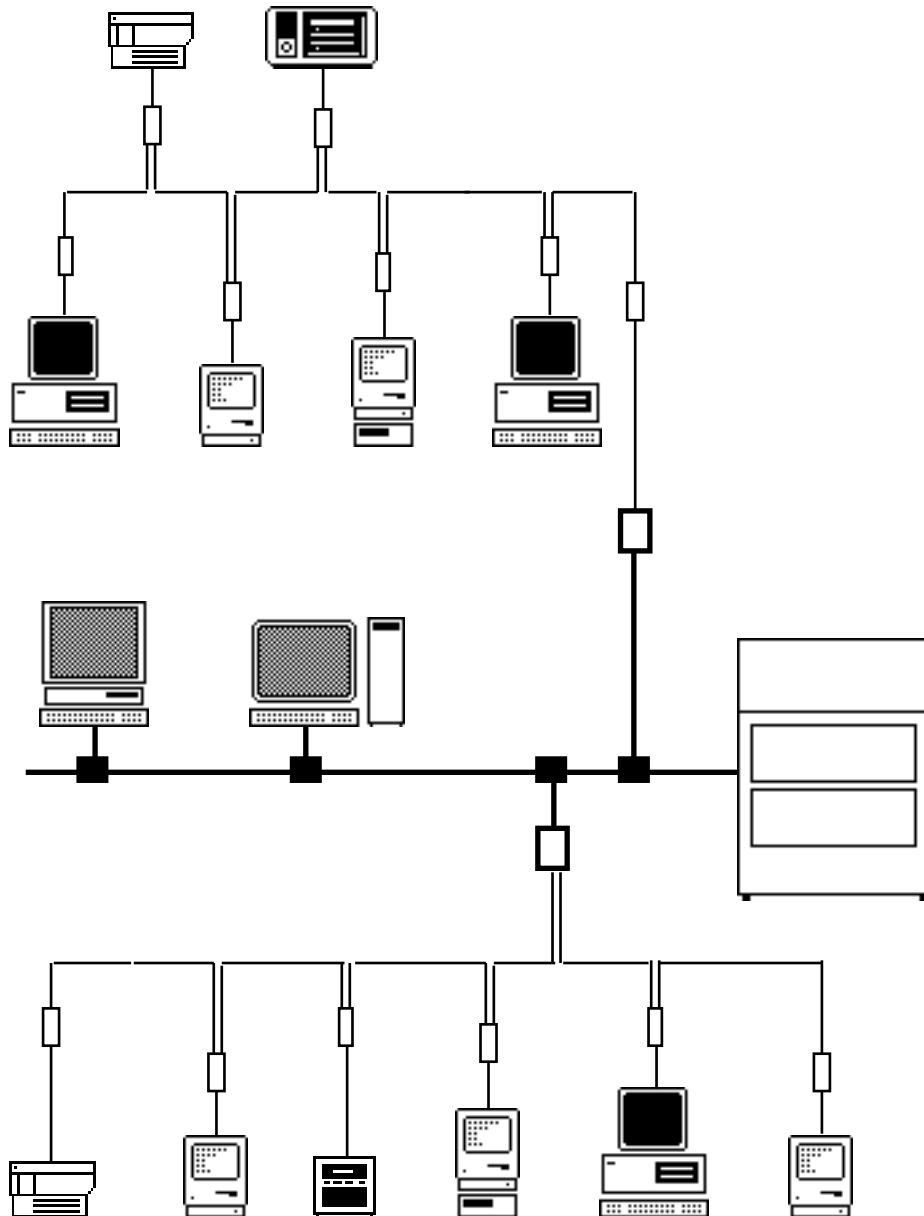


Fig. 11. Det er i dag muligt at opbygge lokale netværkssystemer, der omfatter flere lokale netværk med arbejdsstationer af forskellig type. De forskellige netværk er koblet sammen med "porte", dvs. specialiserede datamater.

Man kan selvsagt ikke integrere alt med alt. Men det er altså i dag muligt at skabe en integrerede netværksløsning omfattende flere netværkstyper forbundet med "porte" og med tilslutning af et meget bredt spektrum af arbejdsstationer som f.eks. IBM's PC, Macintosh, Xerox 6085, med tilslutning af minidatamater som Digital's VAX og med mulighed for at etablere kommunikationsforbindelse fra nettet til centrale værtsdatamater.

I strategisk planlægning kan man derfor som hovedregel gå ud fra, at det er realistisk at integrere forskellige datamater af forskellig fabrikat og endda med forskellige kommunikationsprotokoller. For nogle kombinationers vedkommende vil integrationen selvsagt stadig en overgang volde problemer. Men brugernes vedholdende krav om fleksibilitet med henblik på at kunne integrere sammensatte

systemer medfører, at markedet for porte og protokolprogrammer er hastigt voksende. Som præmis for planlægningen af tekniske organisationers teknologistrategi er der ingen grund til at antage, at det, der gjaldt i går, også vil gælde om to eller fem år.

Udgangspunktet må være, at der til de forskellige opgaver anskaffes netop det system, der giver den bedste støtte til arbejdet, uanset operativsystem, leverandør, etc. Det funktionelt set bedste system kan naturligvis være for dyrt for organisationens budget; i så fald kan man selvsagt blive nødt til at nøjes med det næstbedste system. (Men husk at medtage alle beløb i beregningen - herunder uddannelsesomkostninger, forventet produktivitetsstigning etc.). Det kan naturligvis også vise sig, at det bedste system ikke kan udveksle data med organisationens øvrige systemer. I så fald må man undersøge, om dette problem kan afhjælpes med en "port" mellem det påtænkte system og de øvrige systemer. I de fleste tilfælde vil det vise sig, at problemet kan afhjælpes. Hvis problemet imidlertid ikke kan afhjælpes for nærværende, bør man overveje, om de ulemper, de manglende faciliteter for direkte udveksling af data frembyder, er større end fordelene ved det i øvrigt funktionelt bedste system. Først når dette eventuelt er godtgjort, vil det måske være rationelt at forkaste det funktionelt bedste system. Før man går så vidt, bør man dog endelig undersøge, om en løsning på problemet er forstående, og overveje om man kan leve med problemet indtil da.

Hvem skal opveje fordele og ulemper, vil man måske spørge. Hvem har den fornødne viden til at vurdere, hvad problemet vil betyde for arbejdet, for det tekniske arbejde? Den, der har skoen på, dvs. den der skal udføre arbejdet!

3.4. Åben systemarkitektur

Teknikerens arbejde udviser store forskelle de forskellige brancher, fagområder, udviklingsfaser osv. imellem og er tilmed i bestandig forandring.

Teknikerens arbejdsstation skal derfor kunne konfigureres forskelligt, alt efter karakteren af de opgaverne, den enkelte tekniker skal varetage, og de deraf afledte krav:

- Inputmedier. Til forskellige opgaver svarer forskellige inputmedier. Systemet skal altså frit kunne konfigureres til et stort udbud af inputmedier (mus, stylus, puck). Der vil desuden i nogle kunne være behov for parallelt input; dette stiller særlige krav til operativsystemet.
- Outputmedier. Til forskellige opgaver svarer ligeledes forskellige outputmedier. I nogle sammenhænge er der behov for farveskærm, men ikke i andre. I nogle sammenhænge kan der være behov for farveplotter osv.
- Krav til maskinhukommelse og pladelager. Nogle opgaver (f.eks. simulation) stiller meget store krav til maskinhukommelse.
- Krav til regnekraft. Nogle opgaver (f.eks. 3D CAD, simulation af komplekse systemer) stiller meget store krav til regnekraft, som vil kræve særligt hardware, f.eks. en simulationsaccelerator på lokalnettet.
- Afhængighed af import af viden fra centrale databaser.
- Osv.

Arbejdsstationens åbne arkitektur må ikke være anarkistisk. Den skal ikke blot være åben, men - til forskel fra MS DOS maskinerne - være forberedt for åbenheden. Der må ikke kunne opstå problemer med kompatibilitet mellem hardware

komponenter, mellem disse og applikationsprogrammel, eller mellem arbejdsstationer med forskellig konfiguration.

Forskellene i arbejdets karakter de forskellige kategorier af teknikere og de forskellige afdelinger imellem betyder, at man må forvente, at ikke alle krav kan imødekommes af samme type arbejdsstation eller af samme operativsystem. Man må derfor være indstillet på at skulle arbejde med arbejdsstationer af forskelligt fabrikat og med forskellige styresystemer. Dette indebærer, at de lokale datanet nødvendigvis bliver heterogene. Sådanne heterogene netværksløsninger er i dag under hastig udvikling.

4. Konklusion

Grundlaget for TIA er en eksplicit repræsentation af det tekniske arbejdes beslutningsrum.

TIA skal øge den regnekraft, der står til rådighed for den enkelte beslutningstager såvel som for kollektivet, og vil dermed øge den informationsmængde, der på et givet punkt reelt kan bearbejdes i beslutningsprocessen. Det vil ikke mindst betyde, at det søgerum, der kan overskues, vil kunne være langt større.

TIA skal frembyde faciliteter der direkte understøtter beslutningsprocessen ved at stille et bredt spektrum af faglig viden til rådighed for de enkelte teknikere på de enkelte punkter i beslutningsprocessen og ved at give teknikeren mulighed for interaktivt og meget hurtigt at koncipere en løsning og evaluere dens konsekvenser.

TIA skal reducere forbruget af arbejdskraft til informationsbehandling per iteration, bl.a. ved at gøre det til en opgave af forsvindende omfang at foretage rettelser som følge af ændringer o.l. i form af udarbejdelse af nye konstruktionstegninger, gennemføre nye beregninger, foretage ændringer af rapporter, udarbejde nye styklister, produktionsplaner osv.

TIA skal støtte det tekniske arbejdes iterative karakter ved at gøre iterationerne mindre arbejdskrævende såvel som mindre tidskrævende. Det bliver dermed på den ene side muligt at øge antallet af mulige iterationer, øge antallet af alternative løsninger under overvejelse og dermed øge kvaliteten af produkter eller på den anden side at reducere gennemløbstiden og omkostningerne.

TIA konceptet kan ikke realiseres ved en kosmetisk foreanstaltning på en række eksisterende maskiner og applikationer. TIA konceptet må realiseres ved at tjene som retningslinje for konstruktion af nye systemer, dvs. ved at foreskrive, hvilken funktionalitet sådanne systemer skal kunne frembyde, hvorledes de skal udformes etc.

Litteratur

- Thomas J. Allen: *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization* (1977), The MIT Press, Cambridge, Mass. - London, 1984.
- Christopher Alexander: *Notes on the Synthesis of Form*, Harvard University Press, Cambridge, Mass. 1964.
- M. Myrup Andreassen, H. Stahl og E. Tjalve: *Konstruktionsprocessens faser*, Konstruktionsinstituttet, AMT, DTH, Lyngby 1972.
- P. F. Arthur: "The Engineering workstation: some thought on its future form", *Computer-Aided Design*, vol. 17, nr. 3, april 1985, pp. 115-116.
- Morris Asimow: *Introduction to Design*, Prentice-Hall, Englewood Cliffs, New Jersey, 1962.
- U. Bauenreind: "Realisierung von CIM-Konzepten mit Standardkomponenten", *Zeitschrift für wirtschaftliche Fertigung*, vol. 80, 1985, nr. 9, pp. 397-403.
- Ronald Baecker and William A. S. Buxton (eds.): *Readings in Human-Computer Interaction: A Multidisciplinary Approach*, Morgan Kaufmann Publishers, 1987.
- G. Bechmann, R. Vahrenkamp und B. Wingert: *Mechanisierung geistiger Arbeit. Eine sozialwissenschaftliche Begleituntersuchung zum Rechnereinsatz in der Konstruktion*, Campus Verlag, Frankfurt/NewYork, 1979.
- Llewellyn W. Bezanson m.fl.: "Engineering Support System User Requirements", *IEEE Micro*, oktober 1985, pp. 36-51.
- L. L. Bucciarelli: "Reflective practice in engineering design", *Design Studies*, vol. 5, no. 3, July 1984, pp. 185-190.
- William Buxton and Ronald Baecker: "Human-Computer Interaction: Selected Theories, Technologies, Techniques & Tools", *SIGGRAPH '86*.
- John R. Dixon: *Design Engineering: Inventiveness, Analysis, and Decision Making*, McGraw-Hill, New York etc., 1966.
- Joseph Harrington, Jr.: *Computer Integrated Manufacturing*, New York 1973, reprint: Robert E. Krieger Publ. Co., Malabar, Florida, 1979.
- Joseph Harrington: *Understanding the Manufacturing Process. Key to Successful CAD/CAM Implementation*, Marcel Dekker, New York, 1984.
- D.J. Hartzband and F. J. Maryanski: "Enhancing Knowledge Representation in Engineering Databases", *IEEE Computer*, vol. 18, nr. 9, september 1985, pp. 39-48.
- Les Holland m.fl.: "Engineering Support System Software", *IEEE Micro*, oktober 1985, pp. 17-21.
- Edwin L. Hutchins, James D. Hollan and Donald A. Norman: "Direct Manipulation Interfaces", i Donald A. Normand and Stephen W. Draper (ed.): *User Centered System Design. New Perspectives on Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, New Jersey, - London, 1986, pp. 87-124.
- Harvey Jones: "Creating the integrated engineering design office", *Microelectronics Journal*, vol. 13, nr. 6, 1982, pp. 15-17.
- Raphael Kaplinsky: *Automation, the technology and society*, Longman, Harlow, 1984.
- Donald O. Knight (ed.): *Engineering Joint Study. Engineering Workstation-Workshop, April 26-27, 1984*, College of Engineering and Applied Sciences, Arizona State University, Tempe, Arizona.
- Donald O. Knight (ed.): *Engineering Joint Study. Engineering Workstation-Workshop II, February 7 & 8, 1985*, College of Engineering and Applied Sciences, Arizona State University, Tempe, Arizona.
- Donald O. Knight: "The Engineering Workstation and the Engineering Support System - Present Status, Future Directions", *IEEE Micro*, oktober 1985, pp. 6-9.
- Donald O. Knight: "The Engineering Workstation and the Engineering Support System", TIA Seminar, november 1986.
- Donald O. Knight: "Engineering Workstation User Requirements", TIA Seminar, 1. november 1986.

- John K. Krouse: "Automation revolutionizes mechanical design", *High Technology*, vol. 4, nr. 3, marts 1984, pp. 36-45.
- Mike Merritt m.fl.: "Desires and Aspirations of the Engineering Support System User", *IEEE Micro*, oktober 1985, pp. 10-17.
- Eirik Næss-Ulseth and Trygve Reenskaug: *Effectiveness and Quality in Engineering (EKI) Program Overview, January 1985*, EKI Report no. 2, Senter for Industriforskning, Oslo 1985.
- Dan R. Olsen, William Buxton, Roger Ehrich, David J. Kasik, James R. Rhyne and John Sibert: "A Context for User Interface Management", *IEEE Computer Graphics and Applications*, vol. 4, nr. 12, december 1984, pp. 43-50.
- Dan R. Olsen, Mark Green, K. A. Lantz, A. Schultert, J.L. Sibert: "Whither (or wither) UIMS?", *CHI+GI 1987*, pp. 311-314.
- Klaus W. Otten: "Information Systems for Engineering and Manufacturing: CAE, CAD, CAM, CIM", *Journal of Information & Image Management*, vol. 17, nr. 5, maj 1984, pp. 25-30.
- Jens Rasmussen: *On the Structure of Knowledge - a Morphology of Mental Models in Man-Machine System Context*, RISØ-M-2192, Risø National Laboratory, Roskilde, November 1979.
- Jens Rasmussen: "Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-13, no. 3, May/June 1983, pp. 257-266.
- Jens Rasmussen: "The Role of Hierarchical Knowledge Representation in Decisionmaking and System Management", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-15, no. 2, March/April 1985, pp. 234-243.
- Jens Rasmussen: *Information Processing and Human-Machine Interaction. An Approach to Cognitive Engineering*, North-Holland, New York etc., 1986.
- Jens Rasmussen: "Menneske-maskin systemer. Principper for design af intelligente systemer", Indlæg på FTU-Seminaret, 27.-28. august 1986, DTH, Lundtofte.
- Jens Rasmussen: "A Cognitive Engineering Approach to the Modelling of Decision Making and Its Organization in Process Control, Emergency Management, CAD/CAM, Office Systems, Library Systems", i: W. B. Rouse (ed.): *Advances in Man-Machine Systems Research*, vol. 4, JAI Press, Greenwich, Conn., 1987. (Manus. offentliggjort af Forskningscenter Risø, Juni 1986; Risø-M-2589).
- Philip Robinson: "A World of Workstations", *Byte*, November 1987, pp. 251-261.
- Shyamal Roy: "Get the most Out of Low-Cost CAD", *Designfax*, vol. 8, nr. 7, juli 1986, pp. 35-42.
- Steve Sapiro: "The Electronic Workstation - An Overview", *IEEE Design & Test*, november 1984, pp. 33-41.
- Ben Shneiderman: "Direct Manipulation: A Step Beyond Programming Languages", *Computer*, vol. 16, nr. 8, august 1983, pp. 57-69.
- Thomas W. Sidle: "Weaknesses of Commercial Data Base Management Systems in Engineering Applications", *Proceedings of 17th Design Automation Conference, Minneapolis, June 1980*, ACM, pp. 57-61.
- Herbert A. Simon: "The Structure of Ill Structured Problems", *Artificial Intelligence*, vol. 4, 1973, pp. 181-201.
- Herbert A. Simon: *The Sciences of the Artificial*, 1969, 2nd ed., MIT Press, Cambridge, Mass. - London, 1981.
- Herbert A. Simon: "Search and Reasoning in Problem Solving", *Artificial Intelligence*, vol. 21, 1983, pp. 7-29.
- Hal L. Stern: "Comparison of Window Systems", *Byte*, November 1987, pp. 265-272.
- Eskild Tjalve: *Skematisk udformning af industriprodukter - Værktøjer for konstruktøren*, Akademisk Forlag, København 1976, 2. oplag 1983.
- Undersøgelse omkring Sammenkobling af Administrative EDB systemer og Tekniske EDB systemer*, F. L. Smidth & Co. A/S, Juni 1985.
- John Voelcker, Paul Wallich and Glenn Zorpette: "PC applications: engineering", *IEEE Spectrum*, vol. 23, nr. 5, maj 1986, pp. 68-75.
- Paul Wallich: "A review of engineering workstations", *IEEE Spectrum*, vol. 21, nr. 10, oktober 1984, pp. 48-53.
- Paul Wallich: "Engineering workstations: a field report", *IEEE Spectrum*, vol. 21, nr. 12, pp. 52-56.

Gregg Williams: "The Apple Macintosh Computer", *Byte*, vol. 9, nr. 2, februar 1984, pp. 30-54.
Yalamanchill, M. Malek and J. K. Aggarwal: "Workstations in a Local Area Network Environment", *IEEE Computer*, november 1984, pp. 74-86.