NOMINALSYNTAGME-SPECIFICITET OG DISKURSUNIVERSER

Henrik Prebensen

Humanistisk edb-center
Københavns Universitet

***

1. Datalingvistik og eksperimentel lingvistik

Datamatiske systemer til forståelse af naturligt sprog vil kunne få en række praktiske applikationer, fx som grænseflader i forbindelse med databaser, ekspertsystemer, systemer til planlægning og styring af allokation af ressourcer mv.

Svagheden ved sådanne forståelsessystemer er deres relativt ringe plasticitet, dvs. evne til at klare en naturlig kommunikationssituations mange regelbrud og tvetydigheder. Interessen for dem ligger især i, at de kan spare brugerne af avancerede datasystemer for tid og besvær med at lære særlige formelle kommunikationssprog.

Opgaven er at overvinde svaghederne på en tilfredsstillende måde, så forståelsessystemer kan konkurrere effektivt med ikoniske systemer, menusystemer ol. Studiet af forståelsessystemer er derfor af klar interesse for datalingvistikkens anvendelsesmuligheder.

Dette studium har imidlertid også lingvistisk teoretisk interesse. Ved at implementere lingvistisk viden i sådanne systemer iværksætter man en form for eksperimentel lingvistik. Den kan have samme positive indvirkning på lingvistikkens udvikling, som simulation og eksperimentel kalkyle har haft og har i andre videnskaber.

Implementering af teorier, hypoteser, empirisk viden bidrager til opdagelsen af ny formalismer. Formalisering er ofte en løftestang for ny erkendelse. Implementeringen af regler mv. for konkrete sproglige fænomener giver ydermere mulighed for at evaluere gyldigheden af disse regler og rækkeviden af de begreber, de benytter. Refleksion over systemstruktur giver ligeledes mulighed for ny indsigt i sprogvidenskabelige begrebsdannelser.

Nominalsyntagmespecificitet og diskursuniverser
Henrik Prebensen
Proceedings of NODALIDA 1987, pages 78-95

Med en eksperimentel datamatisk implementering af et grammatisk fænomen igangsættes en dialektisk proces mellem afprøvning af hypoteser, afklaring af begreber, udvidelse af problemstillinger og sammenkobling af teorier og vidensområder, en proces, der kan være af stor gavn for forståelsen af sprog og tænkning i almindelighed. Med datamaten som eksperimentelt værktøj står lingvistikken i en ændret forskningssituation.

2. Blockhead eksperimentet

Dette foredrag bygger på et arbejde med et eksperimentelt lingvistisk forståelsessystem skrevet i PC-PROLOG (alias TURBO-PROLOG). Eksperimenterne vedrører betydning og forståelse af nominalsyntagmer, der refererer til genstande i en ydre virkelighed.

Det eksperimentelle system er en simpel klodsverden, hvori en robot, *Blockhead*, ved hjælp af en krog kan flytte rundt på en lille samling klodser på et bord og besvare spørgsmål om klodsverdenens tilstand. Tilstanden vises i et vindue på en dataskærm. Ordrer og spørgsmål til *Blockhead* skrives i et felt under vinduet, hvor også *Blockheads* svar fremkommer. Se i øvrigt skærmbillederne i appendix.

Der gælder de samme vilkår for *Blockhead* som for alle eksperimenter: begrænsningens kunst er vigtig. Eksperimentet uvedkommende omstændigheder må skrælles bort eller reduceres til det minimale. Det væsentlige er, overskueligt og uden "støj" fra parasitfænomener, at kunne manipulere de forhold, eksperimentet er sat op for at studere: i *Blockhead* nominalsyntagmers referenceforhold.

Blockheads verden omfatter derfor kun 8 genstande. Genstandene har kun 3-4 egenskaber. *Blockheads* sprog er et begrænset engelsk, selv om antallet af sætninger og perioder, robotten kan forstå, er (rekursivt) uendeligt.

Ordforrådet omfatter mindre end 40 indholdsord (verber, nominer, adjektiver) og under 70 "grammatiske" ord (pronominer, præpositioner, adverbier, konjunktioner).

Morfologisk analyse er reduceret mest muligt. Fx kan kun præsens indikativ og imperativ, samt enkelte sammensatte former af verber, fx passiv eller *extended form (-ing)* bruges. Nominalsyntagmer optræder kun i singularis, osv.

Det egentlige hovedformål med programmet er at studere **anafori.** De fænomener, der gives en uddybet behandling, er derfor sådanne som spiller en rolle herfor, nemlig nominernes omfangsbetydning eller ekstension, deres bestemthed, deres bestemmelser (relativsætninger, præpositionsforbindelse, mv.), deres optræden i komplekse perioder med sideordnede og betingede sætninger, i spørgsmål og ordrer.

Det er i et sådant arbejde fristende at gøre programmet mere "intelligent", og dermed mere overbevisende og publikumsvenligt ("sexy"). Komplikationer er imidlertid kun berettigede, hvis de er teoretisk interessante. Hensigten med *Blockhead* er ikke at imponere med hensyn til, hvad en datamat kan fås til at gøre, men at øge lingvistens forståelse af sprogets mekanismer, afklare sprogvidenskabelige begreber, gennemprøve formelle værktøjer.

Blockhead-eksperimentet indgår i et projekt: *Anaforisk resolution,* støttet af *Statens humanistiske forskningsråd* under FTU-bevillingen. Projektet er treårigt (1986-88) og foregår ved *Humanistisk edb-center, Københavns Universitet.*

3. Problemet om specifik reference

Det problem jeg her ønsker at diskutere i relation til *Blockhead*-eksperimentet er problemet om nominalsyntagmers specifikke reference: specificitetsproblemet. Specifikke nominalsyntagmer refererer til identificerbare størrelser (entiteter) i et univers; det gør non-specifikke nominalsyntagmer ikke:

(1) a. Marie vil giftes med Ola
(1) b. Marie vil giftes med en nordmand

I (1)a. har *Ola* specifik betydning. Der findes et individ med dette navn, som et andet individ benævnt *Marie* vil ægte. (1)b. har derimod to betydninger afhængigt af *nordmands* specificitet: der kan som i (1)a. være tale om et specifikt individ, som *Marie* vil ægte. Men der kan også være tale om en type. *Nordmand* beskriver da en egenskab ved en ægtemand som *Marie* kunne ønske sig. Der refereres ikke til et specifikt individ, som findes her og nu.

Ved siden af specifik, non-specifik, anvendes andre (næsten) synomyme betegnelser: *transparent/opaque, de re/de dictu*. Fra Frege kendes også betegnelserne *Bedeutung* og *Sinn*, fra logikken *ekstensional* og *intensional* betydning.

Betydningen af et specifikt nominalsyntagme opfattes i moderne semantik som dets *ekstension* eller begrebsomfang, dvs. den mængde af referenter, syntagmet kan udsiges sandt om i et bestemt, givet univers: den aktuelle verden, tekstens virkelighed, diskursuniverset eller hvad man vil kalde det.

Et non-specifikt nominalsyntagmes betydning ses derimod som *intension* eller begrebsindhold; det kan opfattes som refererende til en proces eller metode eller et sæt betingelser, der kan bruges til at bestemme ekstensionen i et hvilket som helst univers. Dvs. betydningen er et sæt af virtuelle ekstensioner: I (1)b. med non-specifik fortolkning skal prædikatet *nordmand* være sandt om det individ, der i en af de fremtidige (mulige) verdener repræsenterer *Maries* ægtemand.

For at undgå at operere med verdener som stedet for nominalsyntagmers ekstension, hvilket kan give mængdeteoretiske problemer, kan man operere med begrænsede "verdenstilstande" eller mulige situationer. Således kan forskellen mellem (2)a. og b.

(2) a. Præsidenten i Santa Marihuana bliver myrdet i dag
b. Præsidenten i Santa Marihuana bliver myrdet lidt for ofte

beskrives ved at sige, at i (2)a. er ekstensionen dagens præsident i *Santa Marihuana*, mens der i (2)b. itereres over et sæt af tilstande, sådan at *præsident i Santa Marihuana* kan udsiges med sandhed om den relevante referent i hver af tilstandene.

Det specifikke nominalsyntagme har sin ekstension i en her- og nu-tilstand defineret ved et bestemt sæt tids- og stedskoordinater. Det non-specifikke har ikke ekstension i en sådan veldefineret tilstand. Derfor forekommer non-specifikke nominalsyntagmer ved modalverber og andre modale udtryk, ved nægtelse, ved iterative udtryk (herunder visse kvantorer), attitudeudtryk, imperativer og spørgsmål. Fælles for disse er, at de peger bort fra den **aktuelle** situation, mod en eller flere **virtuelle** situationer, der er mulige

(modale udtryk), forestillede (imperativer, attitudeudtryk), ukendte (spørgsmål), eller ikke-eksisterende (negation). Skal der ske en anaforisk genoptagelse af non-specifikke nominalsyntagmer i en diskurs, skal der tales i en virtuel modus. Derfor kan anaforer virke baglæns disambiguerende ved tvetydig specificitet:

(3) a. Marie vil giftes med en nordmand. Han hedder Ola.
b. Marie vil giftes med en nordmand. Han skal være stor, lys og blåøjet.
c. Marie vil giftes med en nordmand. Han skal være stor, lys og blåøjet. Han må gerne hedde Ola. ?* Han står derovre.

I (3)a. disambiguerer *han* syntagmet *en nordmand:* ekstensionen må være i den aktuelle verden, fordi fortsættelsen ikke er markeret som virtuel. I b. er fortsættelsen modalt markeret, så ekstensionen henlægges til en fremtidig tilstand. I c. sker der et brud. *Han* optræder først som i b., derefter som i a. Hvis ekstensionen både skal være i den aktuelle og en virtuel tilstand, må diskursen fortolkes spidsfindigt, fx sådan at den talende kender *Maries* fremtid, men at *Marie* ikke selv kender den.

Hvorledes kan et datamatisk sprogforståelsessystem simulere en sådan forståelse af nominalsyntagmer, dvs. hvordan kan den formaliseres i en teori for fænomener som specificitet?

4. Definition af forståelsessystemer

Et semantiske forståelsessystem, som det i Blockhead anvendte, kan formelt defineres som en modelteoretisk struktur, **S**:

(4) $\mathbf{S} = \langle \mathbf{L}, \mathbf{M}, \mathbf{F}, \mathbf{N}, \mathbf{T} \rangle$

hvor **L** et formelt sprog (det semantiske repræsentationsprog),
M er en mængde af entiteter eller genstande (modellen),
F en afbildning, $\mathbf{F}: \mathbf{L} \rightarrow \mathbf{M} \cup \{0,1\}$, (interpretationsfunktionen),
N en mængde af sætninger og perioder i et naturligt sprog,
T en automat, der beregner en afbildning, $\mathbf{T}: \mathbf{N} \rightarrow \mathbf{L}$.

L, det semantiske repræsentationssprog (den semantiske repræsentation), er defineret over et vokabular, **V**, der består af symboler for *relationer* (prædikater), *funktorer* og *termer.*

Hver *relation* og hver *funktor* har en "aritet", der angiver antallet af dens argumenter.

En *term* er et enkelt symbol eller en streng af symboler. En *term* kan være en *konstant*, en *variabel* eller *et komplekst udtryk indledt af en funktor.*

En *velformet formel* (syntaktisk korrekt udtryk) i **L** er en liste, der som hoved har et relationssymbol og som hale en liste af termer, i antal svarende til relationens aritet. Et sådant udtryk kaldes en *proposition.*

En *proposition* er *grundet,* hvis den kun indeholder konstanter, *ugrundet,* hvis den indeholder blot 1 variabel eller funktor.

M, modellen, er det semantiske repræsentationssprogs domæne, dvs. den mængde af genstande, hvori relationerne og termerne i **L** har ekstension.

Man kan i et datamatisk system forestille sig **M** som en *database*, hvor entiteterne er poster eller stamkort, der hver bærer et index eller et navn som identifikator. De egenskaber ved og de relationer mellem entiteterne, der er relevante i en given applikation, repræsenteres da af statiske databaseprædikater, og en given tilstand af modellen repræsenteres af en mængde af databaseklausuler. En forandring, hvorved modellen dynamisk går fra en tilstand til en anden, manifesteres ved, at en given databaseklausul slettes (fx med *retract*), og en anden indsættes (med *assert*). Forandringer sker i overenstemmelse med regler, der indeholder betingelser for sletning og indsættelse.

F, interpretationsfunktionen eller fortolkeren, er en kompleks afbildning, der har udtryk i **L** som definitionsmængde og ekstensioner i **M** eller {**1,0**},(dvs. {**sand,falsk**}) som billedmængde.

F består af en tilskrivningsfunktion, der tager en term i **L** som argument og har en entitet i **M** som værdi og af en evalueringsfunktion, der tager en grundet proposition fra **L** som argument og undersøger, om den er konsistent med modellens tilstand.

Evalueringen afhænger af propositionens type. Hvis propositionen er en kommando, undersøges det om der findes et par af udsagn, der beskriver modellens tilstand før og efter forandringen, og som respekterer en regel for forandring. Hvis propositionen er et udsagn, undersøges dets sandhedsværdi i forhold til modellen.

En proposition, der er udførbar eller sand, kaldes *modelkonsistent*. **F** evaluerer altså *modelkonsistensen* for propositioner i **L**.

F er implementeret som en tilbagesporende proces, der først instantierer variabler og andre ugrundede argumenter for at frembringe en grundet proposition, hvis modelkonsistens så evalueres. Hvis en instantiering viser sig *modelinkonsistent*, sker der tilbagesporing. Således afprøves alle alternative instantieringsmuligheder, inden modelinkonsistens accepteres.

N er en (i princippet uendelig) mængde af sætninger i et "naturligt" sprog, fx *Blockhead-engelsk*.

T er en transducer, dvs. en automat der "oversætter" sætninger i **N** til de semantiske repræsentationer i **L**.

T omfatter som minimum et leksikon, et sæt syntaktiske dekompositionsregler og et sæt semantiske kompositionsregler. **T** foretager en niveaudelt syntagmatisk analyse af en inputsætning til ord- eller morfem-niveau. **T** finder betydningsrepræsentationerne (i **L**) af ord/morfemer i leksikon. **T** danner syntagmernes betydningsrepræsentationer (i **L**) ud fra konstituenternes betydninger og den syntaksregel, der konstituerer hvert syntagme (cf. Freges kompositionsprincip).

I hver regel i **T** foregår der altså på en gang en syntaktisk analyse af et syntagme og en semantisk syntese af syntagmets betydning på basis af de udanalyserede konstituenters betydninger.

Hvis systemet implementeres i PROLOG, kan **T** benyttes i begge retninger, dvs. **T** kan også tage en streng i **L** som input og syntetisere den streng i **N**, der er det natursproglige svar på et spørgsmål.

Den minimale transducer er imidlertid utilstrækkelig til en rimelig simulering af forståelsen af naturligt sprog. Fx kan betydningen af anaforiske udtryk, såsom pronominer, ikke findes ved hjælp af et leksikon. **T** må derfor udvides med en anaforisk proces, der tillader at gemme og hente sådanne betydningsrepræsentationer, som er afhængige af konteksten.

Endvidere er det hensigtsmæssigt at lade **T** benytte **F** til at teste værdier, der tilskrives ugrundede termer *on the fly,* dvs. mens et syntagme analyseres, men inden hele sætningen er analyseret, for at **T** på denne måde altid hurtigst muligt kan give en *grundet* proposition som output.

En fordel ved denne strategi er, at man ved straks at lede efter en konstant som repræsentation for et nominalsyntagme og teste den for konsistens med modellens tilstand undgår *kombinatorisk eksplosion* beroende på syntaktisk flertydighed af nominalsyntagmer. **T** vil altid søge tidligst muligt at finde én og kun én grundet, modelkonsistent repræsentation af input.

T og **F** kan dele informationer om analysen og den semantiske interpretation ved at skrive eller læse på den samme tavle. Tavlen ændrer ikke systemet formelt. Det ville være muligt at undvære den og i stedet overføre alle de relevante informationer som parametre. Ulempen herved er rent teknisk: lange parameterlister og mange tilfælde, hvor parametrene er tomme.

Den her skitserede implementering er teoretisk tilfredsstillende, fordi den giver et formelt veldefineret indhold til mange semantisk-pragmatiske begreber, bl. a. *anafori* og *specificitet*.

På basis af formalismen kan vi nu definere begrebet *forståelse* i forhold til et system ved at sige, at en sætning *P* i **N** forstås af et forståelsessystem, **S**, hvis og kun hvis **T** kan generere en repræsentation af *P* i **L**.

Hvis **S** ikke forstår *P*, kan det skyldes, at *P* er ikke et tilladt input for **T**, fordi *P* er en ukorrekt sætning, eller fordi *P* ikke vedrører det univers, **S** er konstrueret til, fx *Blockheads* klodsverden. Det kan også være at *P* krænker en præsupposition, fx unicitetspræsuppositionen, der omtales nedenfor. Et system som det her definerede vil være i stand til at informere brugeren om grunden(e) til sådanne forståelsesvanskeligheder.

I det følgende beskrives dele af en konkret udformning af et sådant system med *Blockhead* som eksempel.

5. Den semantiske repræsentation, L

L skal kunne repræsentere de tre fundamentale sproghandlingstyper: ordre, spørgsmål og beskrivelse. I gængs logisk semantik har den deklarative sproghandling altid været anset som den fundamentale. Spørgsmål og ordrer behandles ofte slet ikke. Der er imidlertid mange fordele ved at tage de imperative og interrogative typer som primære, dvs. lade forståelsen af dem være forudsætning for behandlingen af den deklarative.

En velformet formel i **L** er en liste med et prædikat som hoved og en række argumenter som hale. Den repræsenterer en sætning i **L**. Hvilken funktion (imperativ, interrogativ, deklarativ) sætningen har, repræsenteres som dens *modalitet.*

(5)	a.	N: put the white block into the box!	
		L: [move, whiteblock, box]	modalitet(!)
	b.	N: is the white block on the table?	
		L: [stat, whiteblock, table]	modalitet(?)
	c.	N: where is the white block?	
		L: [stat, whiteblock, x]	modalitet(?)
	d.	N: which block is situated in the box?	
		L: [stat, xblock, box]	modalitet(?)
	e.	N: there is a block in the box.	
		L: [stat, ablock, box]	modalitet(.)

move og *stat* er relationer, der beskriver henholdsvis forandring og tilstand.

Argumenterne er termer: *whiteblock, box, table* er konstanter; *x, xblock, ablock* er variabler. a.-b. er derfor grundede, c.-e. ugrundede.

Der er 3 slags variabler: den generelle variabel, *x,* der repræsenterer de "totale" spørgeord *what, which, where;* den typologiserede spørgende variabel, *xblock,* der repræsenterer nominalsyntagme med spørgende determinativ, *which block, what block*; den typologiserede indefinite variabel, *ablock*, der repræsenterer nominalsyntagme med ubestemt determinativ, som *a block, some block, any block.*

Der opereres med flere slags variabler, - modsat hvad der tilfældet i semantikker baseret på deklarativ sprogbrug - af hensyn til den korrekte behandling af spørgsmål. Disse deles som bekendt normalt i *helspørgsmål* og *delspørgsmål.* Helspørgsmål indeholder ikke spørgeord og kan besvares med *ja/nej.* Delspørgsmål indeholder spørgeord og kan ikke besvares med *ja/nej,* men med et syntagme, der indsat på spørgeordets sted verificerer udsagnet: *Hvilken klods står i kassen?* - **Den gule**.

Spørgsmål med ubestemt nominalsyntagme er helspørgsmål, men de har en vis lighed med delspørgsmål. De kan besvares med *ja/nej,* men i tilfældet *ja* oftest suppleret med et svarsyntagme, der erstatter det ubestemte nominalsyntagme, ligesom svarsyntagmet erstatter det spørgende syntagme ved delspørgsmål. Et rent *ja*-svar vil i modsat fald ofte afføde et delspørgsmål for at få den supplerende oplysning: *Står der en klods på bordet? - Ja.* - **Hvilken klods? - Den gule**. Svaret kunne derfor lige så godt lyde: **Ja, den gule.** Disse spørgsmål kan derfor kaldes *partielle helspørgsmål.*

For at der kan generes korrekte svar på alle tre slags spørgsmål, må den semantiske repræsentation kode information om spørgsmålets type. Ved det rene helspørgsmål skal sandhedsværdien af et grundet udsagn bestemmes. Ved de andre spørgsmålstyper, hvor der instantieres en variabel, skal der også gives information tilbage om, hvilken instantiering, der har været brugt til at give værdien *sand.*

At de typologiserede variabler noteres som strenge, er en ren notationskonvention. De kunne være noteret "polsk", med funktorer, fx *block(x).*

(5)a.-b. er grundede. *Blockhead* kontrollerer, om konstanterne er navne på entiteter i **M**, og om relationen er modelkonsistent for dem.

(5)c.-e. er ikke-grundede. Her skal fortolkeren **F** prøve at instantiere de variabler med konstanter, der er navne på entiteter, som gør prædikatet modelkonsistent.

Ved nominalsyntagmer, der er definite eller indefinite beskrivelser *(the block on the hook, a block on the table),* genererer **T** en term med en funktor, *IDENTIFY,* som hoved og en proposition i **L** som hale.

(6) a. N: pick up the block in the box!
L: [move, IDENTIFY stat xblock box, hook] modalitet(!)
b. N: place the box on a block on the table!
L: [move, box, IDENTIFY stat ablock table] modalitet(!)
c. N: find a block which is situated on a block on the table!
L: [regard, IDENTIFY stat ablock IDENTIFY stat ablock table]
modalitet(!)

Propositionen efter funktoren *IDENTIFY* er af praktiske grunde noteret som en streng. Den konverteres af fortolkeren **F** til et spørgsmål, og svaret på spørgsmålet indsættes som konstant i den overordnede proposition.

Altså i (6)a. evaluerer **F** *[stat, xblock, box]* som et spørgsmål. Hvis fx *yellowblock* verificerer propositionen, erstattes termen *IDENTIFY stat xblock box* med *yellowblock*. Tilsidst evalueres sandhedsværdien af den derved fremkomne grundede proposition i modellen. Tilsvarende med (6)b.

I (6)c. er der en rekursiv indlejring af termer med funktoren *IDENTIFY.* **F** vil først evaluere den inderste proposition, *[stat, ablock, table].* Lad os sige, at *whiteblock* verificerer den. Nu evalueres den ydre proposition med *whiteblock* som andet argument: *[stat, ablock, whiteblock].* Lad *yellowblock* verificere den. Sidst evalueres *[regard, yellowblock].*

I **L** er termerne altså konstanter, der er navne på entiteter i **M**, fx *whiteblock, box,* variabler af tre slags: *x, xblock, ablock,* der instantieres af **F**, eller funktorudtryk med *IDENTIFY* som hoved, hvor halen evalueres som spørgsmål.

6. Behandlingen af unikke nominalsyntagmer i T

Transduceren **T**s opgave er at generere de korrekte repræsentationer i **L** for nominalsyntagmerne i **N**. Herunder bruger **T** den strategi, at variabler i propositionelle udtryk tidligst muligt skal erstattes med konstanter. **T** søger altså altid at generere en tilfredsstillende *grundet* repræsentation i **L** af inputsætningerne i **N**.

Desuden arbejder **T** ud fra den strategi, at afsenderen altid har gjort sit bedste for at sige noget meningsfuldt, dvs. at **T** på enhver måde skal prøve at forstå (finde en repræsentation af) det sagte.

Transduceren **T** behandler nominalsyntagmer efter to hovedregler: reglen for *unica* og reglen for *non-unica.*

Begrebet *unicitet* betegner en præsupposition vedrørende et nominalsyntagmes ekstension. Denne præsupposition markerer afsenderen med determinativet. Det *bestemte* determinativ signalerer, at referenten er et *unicum,* enestående i modellen, og det forventes, at modtageren kan bruge denne information til at identificere referenten.

Denne type af *unicitet* kaldes *referentiel unicitet.* Hvis der fx i klodsverdenen er én og kun én kasse, omtales den som *the box.* Hvis der er flere, kan ingen omtales som *the box.* Hvis der er én og kun én klods, der er hvid, kan den omtales som *the white block.* Tilsvarende for *the block on the hook, the block on the table which supports the box,* osv.

Referentiel unicitet har været en del diskuteret i logik i forbindelse med Bertrand Russells *theory of definite descriptions.* Problemet er, om brud på uniciteteten giver meningsløse eller falske udsagn: *the present king of France is bald* versus *the present king of France is not bald.* Russell hævder, på grundlag af sin teoris definition af *bestemthed,* at begge udsagn er falske. I *Blockhead* er de meningsløse.

En fejlagtig præsupposition vil i *Blockhead* blive opdaget af **T.** Hvis afsenderen anvender udtryk som *the block which is situated on the table* (eller *the block on the table*), oversættes det, som vi har set, med *IDENTIFY stat xblock table.* Når spørgsmålet *[stat, xblock, table] (which block is situated on the table?)* evalueres, og der står flere klodser på bordet, kan **T** ikke give et svar. Spørgsmålet præsupponerer nemlig unicitet. Fortolkeren, **F,** konstaterer, at der er mere end én klods, der verificerer udsagnet og kan derfor gøre opmærksom på, at spørgsmålet er stillet med forkerte forudsætninger.

F er altså i stand til at undersøge, om en referentiel unicitet er forudsat ved brug af et delspørgsmål. Denne egenskab bruger **T** på adnominale syntagmer som fx relativsætninger eller præpositionssyntagmer med propositionel værdi.

Foruden referentiel unicitet findes *anaforisk unicitet. The block* kan som ekstension godt have en bestemt klods, selv om der ikke foreligger en unik klods i modellen, nemlig hvis den pågældende klods har været omtalt og genoptages anaforisk: *if there is a block in the box, then pick up* **the block!** På samme måde har bestemte pronominer som *it, this* og *that* anaforiske unica som ekstension.

Reglen for unica aktiverer i *Blockhead* en proces, så snart **T** møder et bestemt determinativ. Processen forsøger at læse så meget af den efterfølgende streng som nødvendigt for at identificere det korteste nominalsyntagme, der har et unicum som ekstension. **T** tester herunder hele tiden for såvel referentiel som anaforisk unicitet.

Denne proces er særdeles betydningsfuld ved afgørelse af syntaktisk flertydighed. I sætningen

(7) Put the block in the box on the block on table!

er der syntaktisk set mulighed for flere afgrænsninger af syntagmerne, fx

(7) a. the block // in the box on the block on the table
(7) b. the block in the box // on the block on the table
(7) c. the block in the box on the block // on the table

Præpositionerne *in* og *on* kan nemlig begge være både verbalafhængige (afhænge af verbalet *put*) og relationer i et udsagn *([stat, xblock, box]).* Flertydigheden undgås, hvis der bruges en entydigt verbalafhængig præposition, som *onto*

(7) d. Put the block in the box onto the block on the table!

Når flertydigheden i de fleste tilfælde alligevel ikke erkendes, kan det skyldes, at referenterne er unikke i situationen, og denne (med bestemt artikel signalerede) unicitet, redder entydigheden. Modtageren kan nemlig benytte den til at løse flertydigheden med

det samme, dvs. når nominalsyntagmernes afgrænsninger undersøges. Forståelsen af et nominalsyntagme er altså delvis "lokal".

I et tilfælde som dette, vil **T** først prøve om *the block* alene opfylder unicitetsbetingelsen, fx er anaforisk unik.

Hvis det ikke er tilfældet, vil **T** forsøge om *the block in the box* har en unik referent.

Hvis det heller ikke er tilfældet, forsøges med syntagmet *the block in the box on the table*.

På denne måde undgår systemet den kombinatoriske eksplosion, som en rent syntaktisk analyse ville give.

Uniciteten gør det også muligt at løse tvetydigheder mellem restriktive og parentetiske relativsætninger:

(8) N: Pick up the pyramid which is on the table!
L1: [move, IDENTIFY stat xpyramid table, hook] (restriktiv)
L2: [move, pyramid, /[stat, pyramid, table]?/, hook] (parentetisk)

Ts unicitetsprocedure undersøger først om *the pyramid* er referentiel eller anaforisk unik.

Hvis ingen af delene er tilfældet (L1), samtidig med at unicitetspræsuppositionen siger, at der skal være en unik referent, må informationen i den efterfølgende del af strengen være "nødvendig" for at identificere unicum. Derfor genereres en *IDENTIFY* struktur, som **F** senere behandler som spørgsmålet: *which pyramid is placed on the table?*. Svaret udgør den eftersøgte unikke referent.

Hvis *the pyramid* er referentiel eller anaforisk unik (L2), kan der ikke i reststrengen være indeholdt information, der er nødvendig for identifikationen af en unik referent. Hvis der derfor optræder en relativsætning, kan den kun være parentetisk. Den behandles derfor som et helspørgsmål: *is the pyramid placed on the table?,* der forventeligt skal besvares med *yes,* hvis ikke relativsætningen skal være nonsens.

Sammenfattende kan det siges, at unicitetsanalysen hviler på muligheden af at bruge såvel morfologisk som syntaktisk, semantisk og pragmatisk information til at afgrænse det bestemte nominalsyntagme lokalt, dvs. uden hensyntagen til det overordnede strukturniveau.

Strategien bygger på, at substantivagmet altid begynder med et bestemt determinativ til venstre. Derefter følger en substantivisk kerne, der evt. kan foregås af et adjektivsyntagme:

(9) $[_{np}$ the $([_{ap}$ small black$])$ $[_{n}$ block$]$ ...

Kernesubstantivet kan syntaktisk set være højreafslutningen på hele syntagmet. Det, der testes ved unicitetsproceduren er, om denne afgrænsning giver mening i konteksten, altså om der findes en unik referent eller anafor til *the (small black) block.* Den semantiske evaluering sker på stedet ved hjælp af fortolkerens spørgemekanisme.

Hvis svaret på fortolkerens spørgsmål er *ja,* må resten være et syntagme med funktion på det højere niveau (verbalafhængigt fx) eller en parentetisk udvidelse til *the (small black) block,* svarende til et indskudt udsagn, dvs. noget som kan evalueres som et helspørgsmål.

Hvis svaret er *nej,* må en del af den efterfølgende streng skulle medinddrages i syntagmet før højregrænsen kan sættes. **T** gnaver sig derfor frem til næste potentielle syntagmegrænse og evaluerer den derved fremkomne syntese i forhold til modellen for at se, om grænsen ligger der:

(10) $[_{np}$ the $([_{ap}$ small black$])$ $[_{n}$ block$]$ $[_{restr}$ in the box$]$ ...

Således fortsættes, så længe der ikke er fundet en meningsfuld substantivisk helhed og der stadig til højre er en streng, der syntaktisk kan være en del af et substantivsyntagme.

7. Behandlingen af non-unikke nominalsyntagmer i T

Hvis et substantivsyntagme er indledt af ubestemt determinativ, er der ikke gjort nogen antagelse om unicitet. Dvs. modtageren er frit stillet i sine fortolkningsmuligheder. Der vil normalt være tale om mange mulige referenter, selv om der godt kan være tale om kun én. Det afgørende er, at afsenderen ikke giver nogen information om forudsætningerne, og at modtageren derfor frit må vælge en referent inden for de givne muligheder.

Når **T** læser et ubestemt nominalsyntagme

(11) $[_{np}$ a $([_{ap}$ black$])$ $[_{n}$ block$]$ ...,

kan **T** imidlertid ikke på basis af en lokal semantisk syntese foretage et valg. Dels kan der i det efterfølgende være udvidelser til syntagmet med informationer, der begrænser valgmulighederne. Dels kan valget være begrænset af kravet om, at hele sætningen eller hele den periode, sætningen indgår i, skal være meningsfuld, dvs. modelkonsistent. **T** genererer derfor en midlertidig repræsentation af syntagmet (en variabel), der tillader at udsætte valget.

Hvis der fx er tale om sætningen

(12) move a block into the box!

nytter det ikke at instantiere med navnet på en klods, der senere viser sig ikke at kunne være i æsken. Derfor genereres variabelen, *ablock*, der senere instantieres af fortolkeren *med tilbagesporing,* indtil der opnås en grundet, modelkonsistent ordre.

Hvis der fx er tale om perioden

(13) pick up a block and put it into the box and place the pyramid on it!

kan det ikke nytte at *a block* instantieres med navnet på en klods, der er for stor til at være i æsken og for lille til, at pyramiden kan stå på den. Valget er begrænset til den delmængde af de disponible klodser, der opfylder de rette krav. Dette problem løses i *Blockhead* med et "fantasi-modul", der er en del af fortolkeren.

Fantasi-modulet opretter en stak af virtuelle tilstande med de forandringer, der svarer til en række kommandoer. På den måde sker der forward-chaining frem mod målet: en instantiering, der gør alle sætningerne i en periode modelkonsistente.

I (13) vil fantasimodulet først gemme den aktuelle tilstand. Nu vil **T** søge en repræsentation til den efterfølgende sætning. Fantasimodulet vil oprette en ny tilstand, der svarer til den beordrede forandring, og stakke den ovenpå den første, og derefter fortsætte således til sidste sætning. Hvis det undervejs viser sig, at en proposition ikke er modelkonsistent med den sidst stakkede tilstand, spores der tilbage gennem stakken til sidste tilstand, i hvilken der fandtes en alternativ instantiering. På denne måde foregår der en søgning igennem et træ af instantieringer, indtil der er fundet en modelkonsistent serie af repræsentationer af kommandoerne, eller indtil alle muligheder er udtømt.

Hvis der er en udvidelse, fx relativsætning, til et ubestemt nominalsyntagme:

(14) a. N: a block which supports the white one

genereres en *IDENTIFY* term svarende til et partielt helspørgsmål:

(14) b. L: IDENTIFY stat-invert ablock whiteblock

F evaluerer (14)b. som spørgsmålet *does any block support the white one?* og indsætter den konstant, der indgår som svar, fx *yes - the big black block* - på *IDENTIFY*-termens plads.

Det afgørende ved den non-unikke analyse af ubestemte nominalsyntagmer er altså, at modtageren kan vælge mellem flere instantieringer af syntagmet med konstanter, men at valget er bundet af betingelser, som sikrer, at den resulterende proposition er modelkonsistent.

Strategien for **T** er at repræsentere non-unikke nominalsyntagmer med variabler, der instantieres tidligst muligt ved kald af **F**, evt. under anvendelse af fantasi-modulet, der gennemprøver alle alternativer.

8. Anaforer

Der opereres med to slags anaforer, *N_anaforer* og *NP_anaforer*.

N_anaforen er det sidst mødte nomen, som transduceren **T** gemmer på systemets tavle. Det benyttes løbende som opslagsord for pronominer som *one* i udtryk som *the white one*.

NP_anaforerne er en liste af de konstanter, der har optrådt som argument-hale i den sidst forudgående proposition. Det er **F**, der sørger for, at de gemmes. De benyttes af **T** til at instantiere pronominer i den næstfølgende sætning.

I det første tilfælde er det en leksikalsk information, der gemmes, i det andet en semantisk. NP_anaforen er ikke (undtagen for så vidt angår genus) afhængig af formen på antecedenten, men af betydningen, af antecedentens extension.

NP_anaforerne er altså extensioner i modellen, som etableres løbende på basis af det sagte. Gives der en ordre

(15) Put the yellow block onto the block in the box on the table!

er anaformængden efter ordrens evaluering entiteterne *yellowblock* og *whiteblock*, dersom *the block in the box on the table* kan instantieres med denne.

9. Diskursunivers og specificitet

Vi kan nu definere begreberne *diskursunivers* og *specificitet* med reference til et modelteoretisk forståelsessystem som *Blockhead.*

> Lad der være givet et modelteoretisk system S.
>
> En sekvens af perioder, $(P_1,...,P_n)$, hvor hver periode består af 1 til m sætninger i N, kaldes *en diskurs.*
>
> Ved en *NP_anafor* i en diskurs vil vi forstå en konstant, der benævner en entitet i modellen, M, og som er forekommet som argument i en proposition, der repræsenterer den sidst forståede sætning i diskursen.
>
> Ved *diskursuniverset* vil vi forstå unionen af modellen og den til enhver tid givne mængde af NP_anaforer.

Herefter kan vi definere begreberne *specifik* og *non-specifik* således:

> Et nominalsyntagme er *specifikt*, hvis det af transduceren, T, repræsenteres med en konstant.
>
> Et nominalsyntagme er endvidere *specifikt*, hvis det af T repræsenteres af en variabel, og af fortolkeren, F, instantieres med en konstant i en modelkonsistent proposition.
>
> Et nominalsyntagme er *non-specifikt*, hvis F ikke kan erstatte det med en konstant.

Disse definitioner bygger på processer, som udføres af et modelteoretisk system, *in casu* systemet *Blockhead.* De kan imidlertid benyttes som eksperimentalteoretiske forklaringer på nogle af egenskaberne ved de tilsvarende fænomener i naturlige sprog. Hermed menes, at disse egenskabers eksistens nu kan udledes som konsekvenser af egenskaber ved systemet, og deres fordeling forudsiges på grundlag af fordelingen i systemet.

For det *første* kan vi nu ud fra systemet forklare, hvorfor bestemte nominalsyntagmer er *specifikke.* Bestemthed er i *Blockhead* implementeret som en proces, der forudsætter at afsenderen ved, hvilken entitet han refererer til. Da han med determinativet markerer en præsupposition om, at referenten er unik i diskursuniverset, følger at han faktisk ved, hvilken entitet han taler om. Den pågældende entitet er derfor specifik *per se.*

For det *andet* kan vi nu ud fra systemet give mening til begerebet *non-specificitet* og forklare den tvetydighed, der er knyttet dertil (se § 3). Et ubestemt nominalsyntagme er *ikke* markeret for *unicitet* og derfor heller *ikke* for *specificitet* fra afsenderens side. Det har

derfor flere tydninger. Det bliver imidlertid specifikt i fortolkeren, **F**, hvis denne finder en ekstension for det i modellen, altså hvis dets betydningsrepræsentation slutter med at indeholde en konstant på det ubestemte syntagmes plads. Hvis derimod fortolkeren ikke kan instantiere det i modellen, er det non-specifikt.

For det *tredie* giver systemet en forklaring på, hvorfor anaforisk genoptagelse kan løse tvetydigheder omkring specificitet. Hvis nemlig et ubestemt nominalsyntagme erstattes af en konstant, gemmes den på tavlen. At der senere sker en vellykket genoptagelse med anafor viser, at der fandtes basis for en sådan på tavlen, og at det ubestemte nominalsyntagme derfor har været instantieret med en konstant, altså været interpreteret specifikt. Hvis vi derimod har en stak af virtuelle tilstande, kan konstanter kun gemmes midlertidigt, nemlig til stakken nedlægges igen. Der er altså ikke bevaret nogen information om instantiering af syntagmet, når den virtuelle modus forlades. Men sålænge **F** opererer i denne modus, er NP_anaforer mulige.

10. Evaluering af systemet

Et modelteoretisk semantisk system som det i *Blockhead* implementerede kan altså benyttes eksperimentelt til at bringe klarhed over en række komplicerede lingvistiske fænomener, syntaktiske, semantiske såvel som pragmatiske.

Metoden hertil er en konkretisering eller anskueliggørelse af et erkendelsesområde beroende på, at der drages analogier til egenskaber og relationer ved elementer i et velforstået formelt system. Denne form for modeldannelse er velkendt i naturvidenskabelige teorier.

Gyldigheden af analogierne og af den forståelse, de fører til, er en kompleks affære. Her skal peges på den særligt eksperimentelle dimension. Ved et virkelighedseksperiment manipulerer man med et begrænset udsnit af virkeligheden, for at kunne drage konklusioner om et andet, evt. blot større domæne. Ved en simulation eller et tankeeksperiment forsøger man at afbilde et symbolsystem på virkelighedsdomænet og prøver at danne billeder af andre tilstande af virkelighedsområdet ved at manipulere med symbolerne. Sådanne simuleringer kan udføres på datamat, hvis de opnår en tilstrækkelig grad af formalisering. Fordelen herved er, som man kender det fra fx meteorologiens modeller, at man meget hurtigt kan overskue langt mere komplekse domæner, relationer osv. end med andre hjælpemidler. Samtidig opnår man en maksimal sikkerhed for, at trivielle "regnefejl" ikke forfalsker resultaterne. Man har selvfølgelig ingen tilsvarende håndfast garanti for, at der ikke er fejl i forudsætningerne, som kan forfalske resultaterne.

Hvis et på datamat implementeret system reagerer i overensstemmelse med vore forventninger inden for et veldefineret testområde, plejer vi at slutte, at systemets regler er isomorfe med lovmæssigheder, der gælder for det studerede område, og at hver udvidelse af systemet, der fortsætter med at opfylde vore forventninger, udvider vor erkendelse af det område, vi slutter analogt til. Systemet fungerer som instrument i en erkendelsesproces.

Blandt de fænomener, *Blockhead* simulerer forståelse af er: bestemthed, komplekse nominalsyntagmers semantik mht. relativkonstruktioner og andre adled, anaforiske relationer i diskurser, specificitet.

Der kan heraf udledes eller verificeres en række grammatiske "love" for disse områder, der forklarer fænomener omkring forståelsen af nominalsyntagmer, deres afgrænsning, referenceforhold, anaforiske egenskaber, anvendelse i relativsætninger mv.

Det har imidlertid også interesse af vurdere mulighederne for udvidelser af systemet og den ny erkendelse, denne forventelig kan kaste af sig.

To mulige udvidelser af dette system forekommer særligt spændende: udvidelse til at omfatte forståelse af *pluralis* og udvidelse til at omfatte forståelse af *berettende tekst*.

Vedrørende *pluralis* er der to veje at gå. Man kan forsøge at behandle nominalsyntagmer i pluralis som udtryk, der refererer til *mængder* af kardinalitet større end 1. En anden løsning, som især er fristende, fordi den umiddelbart tillader at benytte de samme regler i ental og flertal, dvs. at kalde nøjagtigt de samme moduler, kunne bestå i at behandle pluralis *iterativt*. Hermed menes, at udsagn af formen

(16) a. do something with /2/3/.../some/.../all/...entities!
b. do /2/3/.../any/.../all/...entities do something?

behandles som *2, 3, ... , "random", ..., ekshaustive* iterationer over

(17) a. do something with some entity!
b. does any entity do something?

Herved opstår ikke mindst spændende problemer omkring behandlingen af generaliseret kvantificering:

(18) a. every block which is supported by the white block is placed on that block.
b. the block which is supported by the white block is placed on that block.

(18) kan ikke vedrøre en eller flere specifikke klodser. Det er fx sandt uanset om *the block which is supported by the white block* har en referent eller ej. Derfor kan generaliseret flertal ikke bero på iteration, men må bero på en logisk slutningsproces, der evaluerer udsagnet som en tautologi. Dermed opstår behovet for en eller anden form for inferens-modul i fortolkeren.

Vedrørende *berettende tekst* er man i den situation at måtte tage stilling til den videre betydning af begrebet diskursunivers. Som dette optræder i *Blockhead,* er det binært og omfatter dels entiteterne i en database, som man kunne hævde, at der refereres *deiktisk* til, dels NP_anaforerne, som der refereres *anaforisk* til. Der er ikke mulighed for i *Blockhead* at introducere entiteter, hvis eneste "ontologiske basis" er, at de omtales i diskursen. Men det er klart, at beretninger om ikke umiddelbart tilstedeværende fænomener er en så vigtig del af vor sprogbrug, og at det meste af vor viden er baseret på sådanne beretninger: reportage, rapportering, faglitteratur, fiktion osv., at den må med i eksperimentelle undersøgelser.

Der findes forskellige tilløb til løsning af disse problemer i form af såkaldte diskursrepræsentationsteorier. Vanskeligheden ved at give disse teorier tilfredsstillende implementeringer er de mangelfulde semantiske repræsentationsformalismer, der ligger til grund og som oftest bygger på den rene første ordens prædikatskalkyle. Et andet problem rejses med, at beretningers forståelighed ofte beror på encyklopædisk viden, fx om relationer mellem helhed og dele *(the underside of the white block),* ejerforhold *(the owner of the pyramid).* Her overskrides imidlertid ofte grænsen mellem lingvistisk eksperiment på veldefineret grund og forsøg på simulering i stor skala.

11. Kort bibliografi

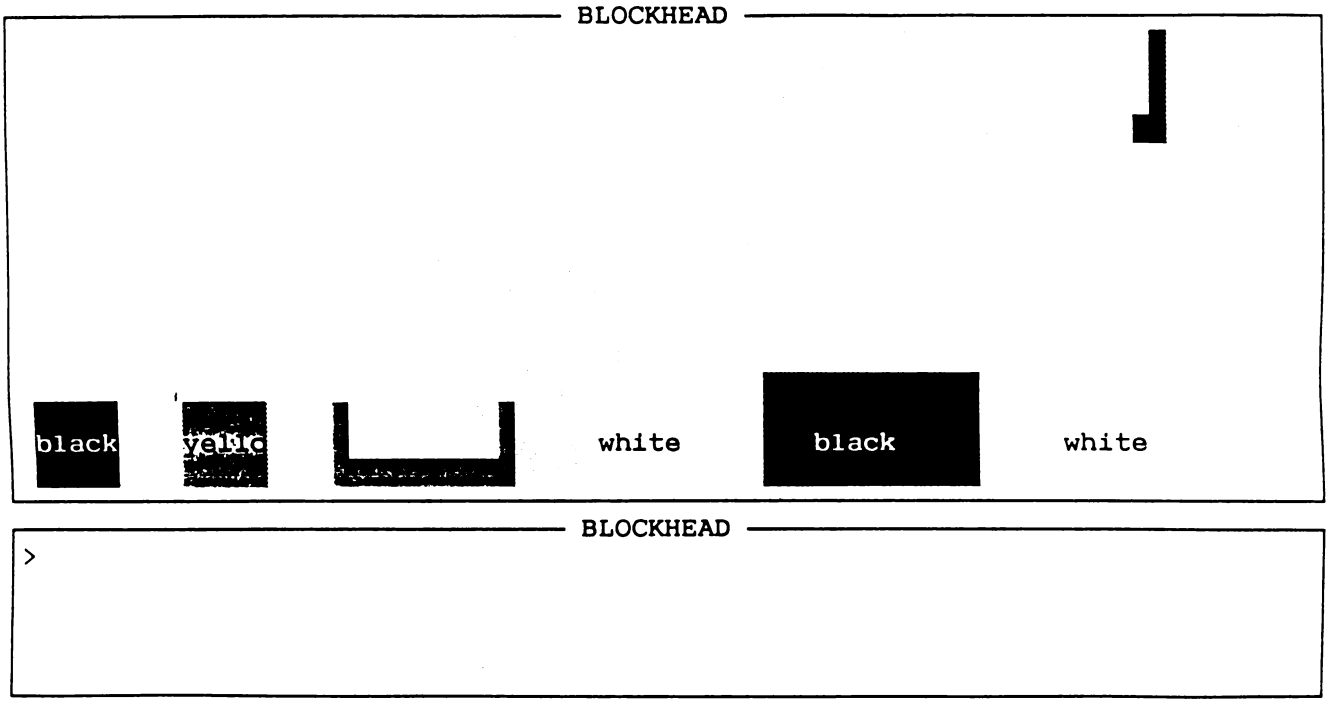
Om modelteori:

J. Allwood, L. Andersson, Ø. Dahl: *Logic in Linguistics,* 1972.
J. Barwise (ed): *Handbook of Mathematical Logic,* 1978.
B. Maegaard, H. Prebensen, C. Vikner: *Matematik og lingvistik,* 1975, kap II.

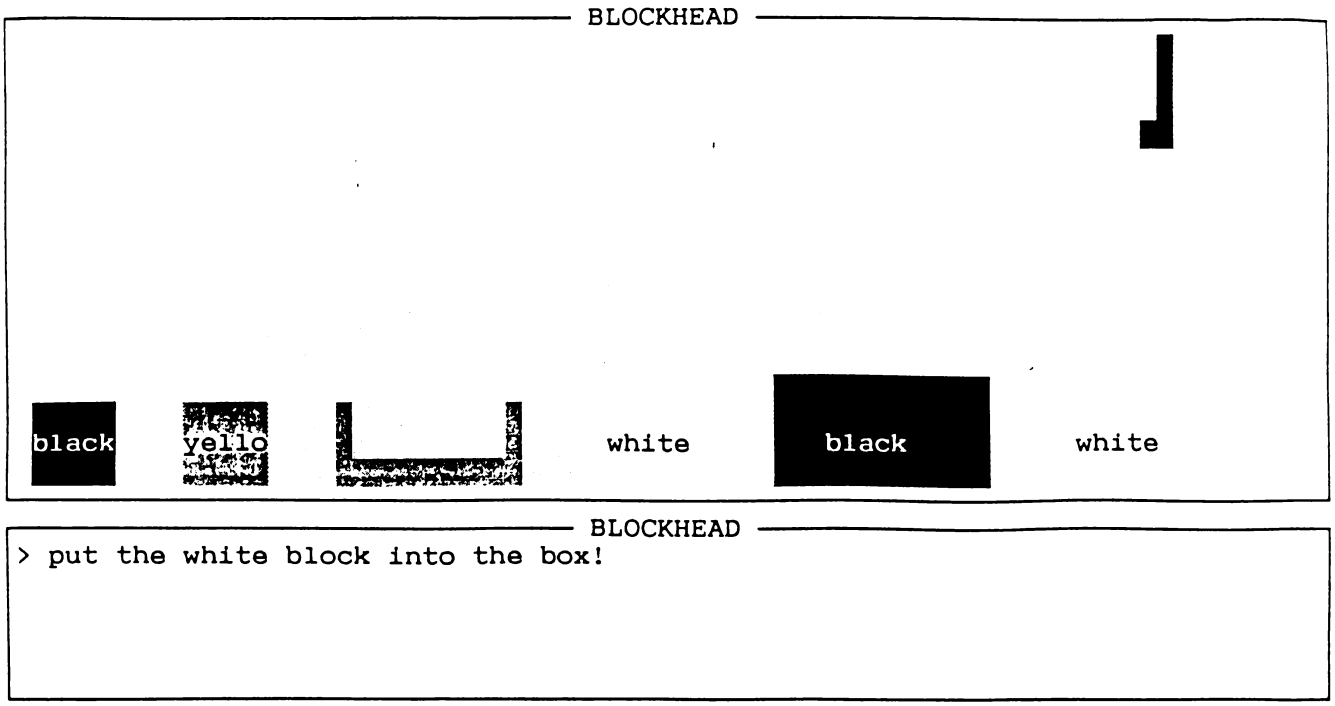
Om anaforer og Blockhead:

Data Humana 4. Humanistisk edb-center. 1987. (Med yderligere bibliografi).
Turbo Prolog. Advanced Guide. Kap. 8 (udkommer i 1988).

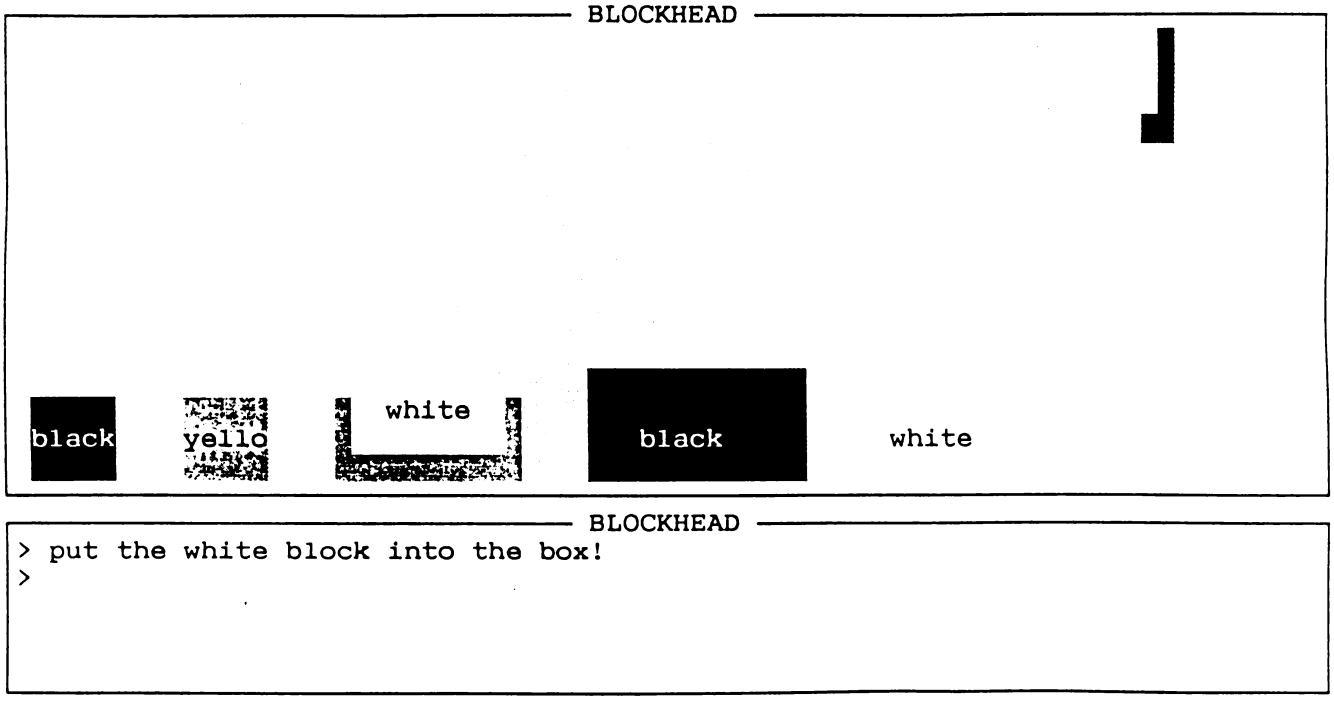
12. Appendiks: Eksempler fra Blockhead-dialog



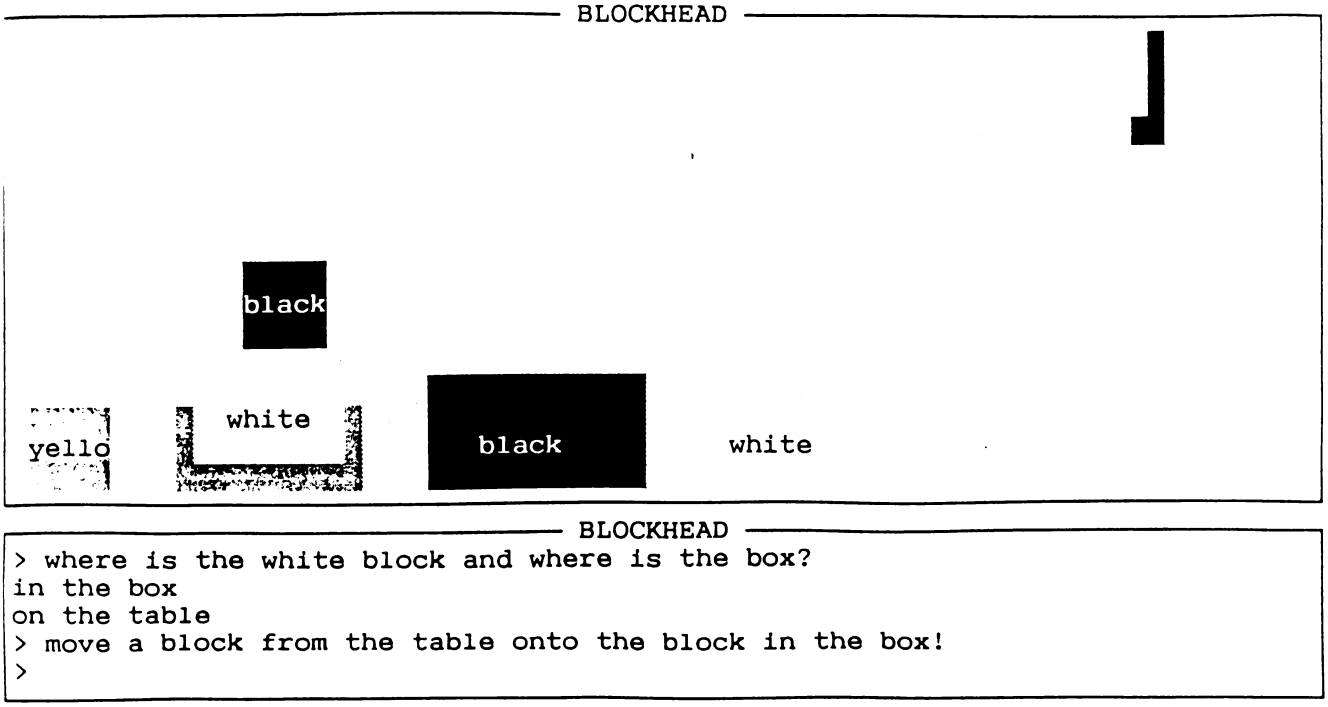
Figur 1: *Blockheads* udgangsposition



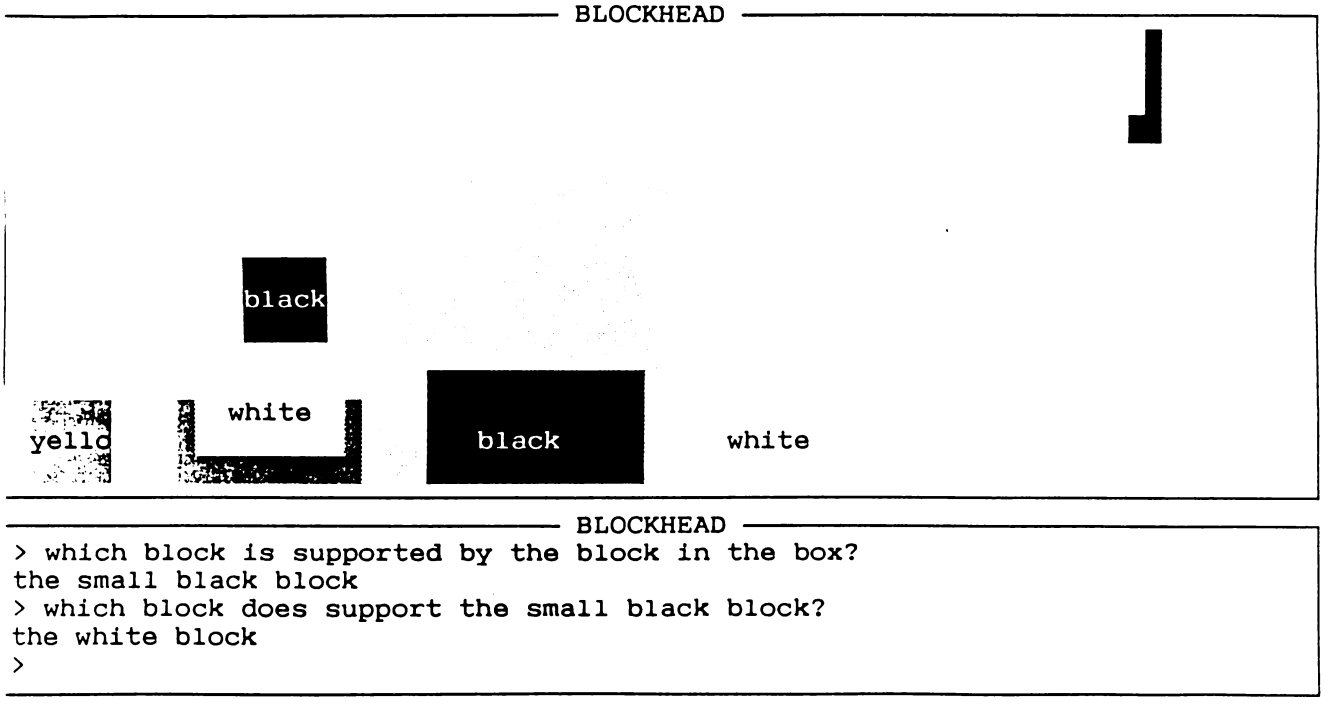
Figur 2: *Blockhead* modtager en en ordre



Figur 3: Ordren udført



Figur 4: *Blockhead* besvarer et spørgsmål



Figur 5.