

Hovedoppgave i
Anvendt Språkvitenskap
okt 2003

Tverrfaglighet og programmering som språkbruk
Et sosio-interaksjonistisk multiagentsystem og ubestemte referanser

av

Ivar Tormod Berg Ørstavik

Institutt for Språk- og kommunikasjonsstudier
NTNU

Forord

Denne hovedoppgaven i anvendt språkvitenskap er skrevet ved Institutt for Språk- og kommunikasjonsstudier, NTNU 2003. Hovedoppgaven er et resultat av et tverrfaglig forskningsprosjekt kalt SjangDAI. SjangDAI ble startet høsten 2000 av Arild Strømhylden og undertegnede, og denne oppgaven markerer slutten på dette prosjektet i sin nåværende form.

Det sentrale problemet i SjangDAI var å *produsere en anvendbar løsning for utvikling av multiagentsystem*. Utvikling av multiagentsystem er et svært sammensatt og omfattende problem som griper dypt inn i både datateknikk og språkvitenskap samtidig. Dette problemet, samt vår anvendte målsetning, skapte derfor en tung problemdrevet, løsningsorientert og tverrfaglig profil i prosjektet, og denne profilen har lagt flere spesielle føringer for denne hovedoppgaven.

Til å begynne med skapte den tunge tverrfaglige profilen en faglig bakgrunn som er unik på mange områder. Dette drev oss til å gå flere språkvitenskapelige og datatekniske forutsetninger, særlig knyttet til teoretiske og metodiske tilnærminger til programmering og programmeringsspråk, etter i sømmene. Arbeidet med utvikling multiagentsystem forutsatte en slik etablert tilnærming, men ettersom ingen etablerte teorier eller metoder kunne gi oss dette, måtte vi etablere en ny underordnet problemstilling: *hvordan fungerer samspillet mellom programmering og programmeringsspråk?* Arbeidet med denne problemstillingen resulterte i en ny teoretisk og metodisk tilnærming, nemlig programmering som språkbruk (kapittel 2).

Underveis i prosjektet oppdaget vi videre at flere underforståtte premisser for tverrfaglig forskning stemte dårlig overens med våre erfaringer og problem som tverrfaglige forskere. Dette drev oss til å etterprøve enda flere tause forutsetninger om for forskning generelt og for tverrfaglig forskning spesielt, og vi måtte derfor etablere enda en underordnet problemstilling: *hva er tverrfaglighet?* Arbeidet i vårt tverrfaglige prosjekt forutsatte en tverrfaglig tilnærming, men ettersom premissene rundt prosjektet stemte dårlig overens med våre behov i prosjektet, måtte vi derfor etablere en egen tverrfaglig plattform og forskerrolle. Denne presenterer jeg i kapittel 1.

Først etter at forutsetningene tilknyttet programmering, språkbruk og tverrfaglig forskning var på plass, kunne vi returnere til vår opprinnelige problemstilling: *hvordan utvikle multiagentsystem*. Denne problemstillingen forutsatte en etablert og fungerende tilnærming til programmering, språkbruk og tverrfaglighet, men disse metodiske, teoretiske og vitenskapsteoretiske tilnærmingen hadde begrenset overføringsverdi for problemene og løsningene i multiagentsystem. I tillegg til dette påvirket også vår uttalte anvendte ambisjon vår analyse og resultat i en tung datateknisk retning. Tunge datatekniske forutsetninger har måttet følge analysene og

resultatene inn i denne oppgaven, og kombinert med mangelen av overføringsverdi fra de to foregående problemstillingene, trengte også det tredje og siste kapittelet å ta tak i sin egen problemstilling ved rota (se kapittel 3).

For å takle disse føringene har denne oppgaven fått en uvanlig form. I stedet for å ta utgangspunkt i en tradisjonell disposisjon med innledning, teori, metode, analyse, resultat og diskusjon, har jeg delt oppgaven inn i tre kapitler. Kapittel 1 tar opp tverrfaglighet og sentrale utfordringer tverrfaglige forskere møter. Dette kapittelet gjenspeiler vår tverrfaglige forskningsplattform og introduserer samtidig noen av de overordnede problemstillingene i prosjektet. Kapittel 2 handler om programmering og språkbruk. Her presenterer jeg et nytt tverrfaglig perspektiv som omfatter både teori og metode fra flere fag. Dette kapittelet forutsetter dermed den tverrfaglige tilnærmingen etablert i kapittel 1. Til slutt i kapittel 3 vil jeg presentere vår tilnærming til multiagentsystem og de programmeringsspråklige ressursene for utvikling av multiagentsystem som vi forsket fram. Denne presentasjonen dreier seg om utviklingsprosessen bak disse løsningene og synliggjør dermed samtidig den kvalitative, datatekniske og anvendte empirien, analysen og resultatene i prosjektet. Kapittel 3 baserer seg på både vårt syn på programmering og språkbruk etablert i kapittel 2 og vår tverrfaglige tilnærming etablert i kapittel 1.

Kapitlene bygger i utstrakt grad på hverandre, og i oppgaven som helhet kan det første kapittelet forstås som innledning, det andre som teori og metode og det tredje som analyse og resultat. Men en for helhetlig tilnærming til denne oppgaven vil også bli misvisende ettersom denne oppgaven ikke er disponert etter en slik mal. Jeg anbefaler lesere av denne oppgaven å se hvert kapittel som en selvstendig tekst som i vekslende grad inneholder både en egen innledning, problemstilling, teori, metode, empiri, analyse og diskusjon.

Jeg vil til slutt rette en stor takk til Arild Strømhylden for et fantastisk tverrfaglig samarbeid. Det var en sann glede og svært lærerikt å jobbe sammen med ham, og resultatene i denne oppgaven er et direkte resultat av vårt samarbeid. Jeg vil også gjerne takke veileder Lars Sigfred Evensen. Det meste jeg kan om anvendt språkvitenskap har jeg lært gjennom ham, enten i forkant av eller under hovedfagsoppgaven, og jeg er stolt av å kalle meg selv en anvendt språkviter. Jeg vil også gjerne takke Agnar Aamodt og mine medstudenter på hovedfag. Responsen de har gitt meg på oppgaven har både gledet, motivert og hjulpet meg svært mye underveis, og jeg har aldri blitt lest bedre. Til slutt vil jeg gjerne takke min kjæreste Randi for alt som ikke har med denne oppgaven å gjøre, og hennes støtte og tålmodighet for alt som har med denne oppgaven å gjøre.

Ivar Tormod Berg Ørstavik

Kapittel 1	Hva innebærer tverrfaglig forskning?	7
1.1	Introduksjon	7
	"Tau kunnen"	7
	Tau kunnen og tverrfaglige forutsetninger	7
1.2	Tradisjonsdrevet, disiplinorientert eller "tverrfaglighet"	8
	Faglighet som interne standarder	8
	Faglighet innenfor etablerte, spesialiserte fagområder	9
	Disiplinorientert eller "tverrfaglighet"?	11
1.3	Tverrfaglig eller disiplinorientert? Ja takk, begge deler!	12
	Tverrfaglig diskusjon, men disiplinorientert faglig arbeid	13
	Flerfaglig samarbeid	14
	Flerfaglig samarbeid som overordnet prosjektramme	14
	Flerfaglig samarbeid som forstadium for disiplinorientert arbeid	15
	Problemer med en flerfaglig fase i et disiplinorientert prosjekt	16
	Hva med flerfaglig utvikling?	18
1.4	Tverrfaglig eller disiplinorientert? "tverrfaglig", takk!	19
	Tverrfaglig samarbeid	19
	Tverrfaglig arbeid krever utadvendthet og flerfaglig læring	19
	Tverrfaglig arbeid krever tilpasning av faglige perspektiv	20
	Etablering av en historieløs, men spesialisert faglig plattform	20
	En "tverrfaglig", historieløs og spesialisert faglig plattform	22
	Nye tverrfaglige plattformers store kommunikasjonsproblem	23
	Nystrand del 1: Tverrfaglige skrivere, disiplinorienterte lesere	24
	Nystrand del 2: tverrfaglig fokus i teksten	26
	Oppsummering av det tverrfaglige kommunikasjonsproblem	27
1.5	SjangDAI - et "tverrfaglig" prosjekt	28
	SjangDAI - En problemorientert, flerfaglig samarbeidsgruppe	28
	Tverrfaglig utadvendthet og læring på tvers av datateknikk og språkvitenskap	29
	Tilpasning av datatekniske og språkvitenskapelige perspektiv	31
	Etablering av en ny, felles, tverrfaglig og spesialisert plattform	34
	SjangDAIs tverrfaglige kommunikasjon	36
	Den tverrfaglige skriverens premisser	38
1.6	Konklusjon: løsning på tverrfaglige kommunikasjonsproblem .	40
	Tverrfaglig lesing	40
	Konsekvenser for resten av denne oppgaveteksten	40

Kapittel 2	Programmering som språkbruk	42
2.1	Samspeilet mellom programmering og programmeringsspråk..	42
	Programmering og programmeringsspråk	42
	Programmering som språkbruk	43
	Språklige forståelsesmodeller overført til programmering	44
	Plan for dette kapitlet.....	45
2.2	En systemisk tilnærming til språk og programmering	46
	En systemisk forståelse av språk	46
	Fordeler og ulemper ved en systemisk tilnærming.....	47
	En systemisk forståelse av programmeringsspråk	48
	Systemene i programmeringsspråk.....	49
2.3	Skygger fra en systemisk tilnærming til programmering	51
	Kreftene bak matematiske system	51
	Konsekvenser av heterogene geometriske system	52
	Kreftene bak programmeringsspråk	53
	Behovet for en ny tilnærming til programmering.....	54
2.4	En dialogisk tilnærming til språk.....	54
	Språk som ytringer, stemmer og heteroglossia	55
	Språkbruk som polyfoni og refraktering.....	56
	Språkbruk som dobbel dialog	57
	Stemmer som sosiale skapte ressurser: grammatikalisering	58
	Fordeler og ulemper ved en dialogisk tilnærming	60
2.5	En dialogisk tilnærming til programmering	61
	Bruk og utvikling av konvensjonelle stemmer i programmering	62
	Bruk og utvikling av leksikalske stemmer i programmering	63
	Bruk og utvikling av grammatiske stemmer i programmering	65
	Heteroglossia i programmering.....	67

[Dette kapittelet er klausulbelagt.]

Kapittel 3	Et sosio-interaksjonistisk multiagentsystem.....	70
3.1	Programmering og multiagentsystem	70
	Objektorientert programmering som språklig drivkraft.....	71
	Objekter i Java som rammer for programmereres perspektiv.....	72
	Hvordan påvirker bestemte referanser programmereres perspektiv?	73
	Situasjonelle problem skapt av bestemte referanser.....	76
	Situasjonelle problem med å styre interaksjon design-time	78
	Behovet for agenter, interaksjon styrt run-time og multiagentsystem	81
	Multiagentsystems muligheter.....	82
3.2	Multiagentsystem, en mauttue og SIMAS.....	83
	Hvilke objektorienterte ressurser finnes og hvilke mangler?.....	83
	Hvordan kan agenter styre sin egen interaksjon og samarbeid?	84
	Anvendt språkvitenskap som programmeringsspråklig ressurs.....	89
	Et SIMAS-perspektiv og utfordringer knyttet til dets realisering.....	92
	Interaksjon og kommunikasjon i SIMAS	93
3.3	"the" SIMAS-agent og ubestemte referanser	96
	Leksikalske elementer i en SIMAS-agent.....	96
	Koblingen mellom kommunikasjon og interaksjon i agent perspective	102
	Det situasjonelle behovet for ubestemte referanser	106
	Bruk av leksikalske ressurser i utviklingen av "Hello".....	107
	Bruk av grammatiske ressurser i "Hello"	109
	Eksempel på SIMAS: to agenter som sier "hello"	112
	Programmereres oppgave.....	114
Referanser	1170

"The curious thing is that we have no clear knowledge of what our presuppositions are and when we try to formulate them they appear quite unconvincing"

(Polanyi om forskning og vitenskap 1962: 59).

Hva innebærer tverrfaglig forskning? Kapittel 1

1.1 Introduksjon

"Taus kunnen"

Polanyi (1962: 264) beskriver de underliggende forutsetningene for forskeres virksomhet som "taus kunnen". Ved å bruke nettopp dette begrepet framhevet Polanyi to egenskaper ved forskning. For det første henspiller "kunnen" på at forskning har en sosial og kulturell omgivelse og at denne omgivelsen blir virksom gjennom den enkelte forskers handlinger. For det andre henspiller "taus" på at denne påvirkningen som oftest ikke diskuteres forskere imellom og dermed gjerne skjer ubevisst hos den enkelte forskeren.

Basert på denne forståelsen av taus kunnen, beskriver Polanyi hvordan dette underforståtte grunnlaget påvirker forskeren og forskningens utvikling. I følge Polanyi åpner ikke de tause handlingsrammene for at forskning kan utfolde seg fritt, men dirigerer retningen som forskning kan utvikle seg innenfor. Taus kunnen blir dermed ikke bare en positiv, muliggjørende drivkraft bak forskning; taus kunnen blir også en negativ, begrensende og usynlig kraft som hindrer forskeren i å søke kunnskap utenfor etablerte handlingsrammer (Polanyi 1962).

Taus kunnen og tverrfaglige forutsetninger

Som forskere bruker vi faglige metoder og verktøy for å forstå et faglig problem, og vi snakker med og tenker i fagspråk. Faglige perspektiv rammer inn våre handlinger, og forskning kan forstås som faglige aktiviteter. Men selv om "faglighet" gjennomsyrrer forskning, er "faglige" egenskaper ved våre handlingsrammer lite omtalt. Det "faglige" brukes derimot gjerne som en betegnelse for underforståtte sider ved våre handlinger som det er vanskelig å fange med ord. Det "faglige" opptrer altså ofte som taus kunnen og er en betegnelse på underforståtte forutsetninger som påvirker vår forskning.

Gjennom å blande faglige verktøy, problemer og språk fra forskjellige fagområder, går tverrfaglig forskning på tvers av etablerte tradisjoner fra disse fagområdene. Som navnet tilsier, bryter tverrfaglige aktiviteter med etablerte faglige handlingsrammer. På denne måten kommer tverrfaglige forskere lett i et motsetningsforhold til etablerte

forutsetninger som inntil da har vært tatt for gitt, og tvinges dermed også til å gå på tvers av den tause kunnen i de etablerte fagområdene (Sørensen 2002, Evensen 2002a, Dahl 1997, Mathisen 1997).

Parallelt med dette motsetningsforholdet mellom taus kunnen i faglig og tverrfaglig forskning, er det likevel et nært slektskap mellom de to forskningsretningene. Bl. a. er begrepet ”tverrfaglig” basert på begrepet ”faglig”, og slektskapet mellom de to begrepene overføres gjerne til også å gjelde taus kunnen og underforståtte forutsetninger. Dette fører til at når vi ikke har erfaringer fra tverrfaglig forskning, baserer vi oss derimot på forskningstradisjoner vi er fortrolig med og innfører taus kunnen fra etablerte fagretninger. Etablerte faglige forutsetninger gjelder dermed ofte som ”default” forutsetninger for tverrfaglig forskning. I tverrfaglig forskning brukes dermed ”faglige” og ”tverrfaglige” forutsetninger om hverandre, og skillet mellom disse handlingsrammene kan virke tydelige i det ene øyeblikket og vage og unødvendige i det neste. Dette utgjør enda et problem med taus kunnen i tverrfaglig forskning.

En forståelse av hva tverrfaglig forskning innebærer har vært en nødvendig forutsetning for vårt prosjektet. Vi ønsket å tverrfaglig kombinere teorier og metoder for å løse et praktisk problem om utvikling av multiagentsystem, og dette skapte en tverrfaglig bakgrunn i prosjektet. Tilknyttet denne plattformen møtte vi noen store utfordringer, og før jeg kan ta for meg min tilnærming til prosjektets hovedproblem, nemlig *utvikling av multiagentsystem*, trenger jeg først å gripe fatt i disse utfordringene tilknyttet tverrfaglighet.

I dette kapittelet vil jeg derfor sette ord på noen former for taus kunnen og underforståtte, ”faglige” forutsetninger i academia. Denne redegjøringen vil jeg bruke til å belyse *hva tverrfaglig forskning innebærer* og diskutere tverrfaglige forutsetninger sett i forhold til etablerte faglige forutsetninger. Først vil jeg derfor ta for meg faglighet som tradisjonsdrevet og disiplinorientert. Deretter vil jeg undersøke hvordan disse forutsetningene fungerer som omgivelse for tverrfaglig forskning. Videre vil jeg beskrive noen særegne tverrfaglige forutsetninger og begrunne disse i mitt eget hovedfagsprosjekt. Til slutt vil jeg vise hvordan slike særegne forutsetninger også må påvirke utformingen av denne oppgaveteksten.

1.2 Tradisjonsdrevet, disiplinorientert eller ”tverr faglighet”

Faglighet som interne standarder

En grunnleggende egenskap ved faglighet slik det forstås i academia, er at faglige perspektiv skal brukes i tråd med etablerte faglige forutsetninger. ”Forskning er primært forpliktet til kravet om sannhet og av de interne faglige standarder som er

utviklet innen forskerfellesskapet” (NESH 2001: Innledning). ”Frihet til å søke og formidle ny kunnskap” (ibid.) er sideordnet kravet om å følge etablerte, faglige konvensjoner, og det samlede bildet blir dermed at søken etter ny kunnskap bør skje i henhold til etablerte faglige interesser. Forskning og kunnskapsutvikling skal altså skje innenfor etablerte, faglige standarder, og dette skaper en spenning mellom å strekke seg mot det nye og forbli ved det gamle.

Et sentralt problem ved dette spenningsforholdet er at lydigheten til interne faglige standarder ofte tas for gitt av oss akademikere. Både i intern og ekstern kommunikasjon vektlegger vi at vi søker ny kunnskap, men samtidig underkommuniserer vi at denne kunnskapsutviklingen må skje innenfor etablerte faglige rammer. Akademiets forståelse av faglighet rommer bare unntaksvis kunnskapsutvikling som skjer utenfor etablerte, faglige rammer. For andre former for nyskaping kan denne betingelsen for nyskaping danne et gjerde som ekskluderer dem fra de akademiske arenaene.¹

Akademiets *tradisjonsdrevne* forståelse av faglighet legger altså opp til at bruk av faglig perspektiv følger etablerte forutsetninger. Men en slik forståelse av faglighet behøver ikke å medføre en kritikk av akademien. En tradisjonsdrevet holdning til nyskaping og faglighet legger svært mange nyttige føringer for søken etter kunnskap og faglig utvikling (jf. NESH 2001, Haavind 2000). Denne presiseringen av ”faglighet” synliggjør bare hvilke forutsetninger som gjelder for forskning. Og ettersom akademien tar mål av seg å også romme tverrfaglig forskning, berører dermed denne forståelsen av faglighet like mye tverrfaglig forskning som annen, mer tradisjonstro forskning.

For tverrfaglig forskning kan forutsetningen om at kunnskapsutvikling skal skje innenfor rammene av ”interne faglige standarder” lett bli en tvangstrøye. Tverrfaglig forskning blander faglig perspektiv på nye måter, og dette kan føre til at deres forutsetninger blir forandret. For tverrfaglig forskning vil dermed etablerte, ”interne faglige standarder” lett kunne komme i konflikt med hverandre og forskningsresultatene, og dette vil kunne forsterke spenningen mellom etablerte faglige standarder og det enkelte forskningsprosjektets faglige resultater. Dersom disse svingningene blir for store, vil dette kunne få store konsekvenser for samarbeidet og gjennomførbarheten i prosjektet.

Faglighet innenfor etablerte, spesialiserte fagområder

En annen og minst like grunnleggende egenskap ved faglighet, er at forutsetningene for og bruken av faglig perspektiv skal deles med og derunder forstås av andre

¹ Hvor er f. eks. alle oppfinnerne på universitetet? Dette er et tankevekkende spørsmål, men for omfattende til å bli diskutert her.

fagpersoner. Dette kravet om at faglig forståelse og resultat skal deles, stammer ikke bare fra akademias kollektive ”søken etter kunnskap for dens egen skyld” (NESH 2002: A1). Faglige perspektiv skal også deles med andre forskere slik at de i sin tur kan vurdere og bygge videre på dem. Akademisk kunnskapsutvikling er i dag først og fremst et kollektivt foretagende. Kommunikasjon er et faglig krav, og om forskning heter det derfor ”publish or perish”.

Men for å kunne kommunisere sine resultater og bruk av faglige perspektiv, trenger forskere å kunne basere seg på en plattform av grunnleggende, gjensidig forståelse (jf. tverrfaglige kommunikasjonsproblem s. 23ff). En slik plattform kan konstrueres på to måter. Enten så må bruken av faglige perspektiv kunne forstås på et generelt grunnlag, et ”generalistregime”, der forskere har tilgjengelig en generell plattform av gjensidig faglig forståelse de kan kommunisere seg imellom på (jf. begrepet ”universitas”). Eller så må det finnes grupper med spesialkompetanse som kan forstå bruken av de faglige perspektivene, et ”spesialistregime”, der forskere har tilgjengelig mange små spesialfaglige plattformer av slik gjensidig forståelse (Sørensen 2002: 7).

Dagens akademiske virkelighet er et spesialistregime. Fagområder har utviklet forskjellige komplekse faglige perspektiv uavhengig av tilsvarende utvikling innenfor andre spesialområder. Forskjellige perspektiv fra forskjellige fagområder kan derfor lett stå i et motsetningsforhold til hverandre, og faglige standarder er derfor interne for de spesialiserte fagplattformene. For å forstå det som skjer på slike spesialplattformer, trenger forskere gjerne mange år med spesialutdanning og sosialisering. Det er derfor ingenting som skulle tilsi at f. eks. en spesialist trent i anvendt språkvitenskap kan forstå den indre faglogikken i f. eks. datateknikk uten flere års ekstra spesialtrening (op. cit.).

I praksis blir derfor de spesialiserte fagplattformene uopnåelige for ikke-spesialister på området. Men hvorfor har så fagområdene spesialisert seg så mye at de har utviklet slike uopnåelige plattformer? Denne spesialiseringen skyldes først og fremst ”den økende kompleksiteten i de problemer som i dag forsøkes løst, vitenskapelig og praktisk, et større ambisjonsnivå og økte krav til løsningenes kvalitet” (Sørensen 1997: 25). Det økte kompleksitetsnivået krever mer presise og avanserte faglige verktøy, og det er bare gjennom å bruke slike faglige perspektiv, at spesialistene greier å håndtere de kompliserte problemene. Spesialiseringen er derfor ikke en luksus fagmiljøene bevilger seg selv for å unndra seg ytre vurdering av sin virksomhet; spesialiseringen er en konsekvens av at kompetansen hos fagpersonene må øke i takt med kompleksiteten på de faglige problemene og perspektivene de må håndtere.

Spredning av faglig forståelse forutsetter at vi har en spesialplattform - av og med fagfeller - som vi kan kommunisere faglig på. Vi er dermed avhengige av at det finnes

likesinnede spesialister som er trent i bruken av de samme faglige perspektivene som oss. Og ettersom kommunikasjon er et faglig krav, får disse plattformene også en gravitasjonseffekt på den omkringliggende forskningen. Faglig kommunikasjon blir trukket mot de etablerte spesialplattformene, og faglighet blir dermed ikke bare tradisjonsdrevet, men også rettet mot de enkelte fagretningene eller *disiplinorientert*.

For tverrfaglig forskning kan en slik disiplinorientering være problematisk. Når tverrfaglige forskere blander faglige perspektiv fra flere fagretninger, risikerer de samtidig å endre på forutsetningene for og bruken av disse perspektivene. Slike endringer vil også påvirke de interne faglige standardene, og den tverrfaglige tilpasningen kan dermed tvinge fram en ny plattform av interne faglige standarder som ligger på siden av etablerte spesialplattformer. Utviklingen av anvendt språkvitenskap som spesialfelt er et eksempel på dette (Evensen 1986). Slik tverrfaglig utvikling av nye spesialplattformer kan være like lite lystbetont og luksuriøs som spesialiseringen innenfor de etablerte fagretningene, og slik tverrfaglig spesialisering er hovedsakelig også problem-, kvalitets- og løsningsdrevet.

Spesialister som er trent i å bruke faglige perspektiv innenfor etablerte fagretninger, vil dermed kunne få store problemer med å forstå de samme perspektivene når de brukes under helt andre forutsetninger på en fremmed, tverrfaglig plattform. Og når den tverrfaglige forskeren mister sine medspesialister, mister han også den ytre plattformen av gjensidig forståelse. Uten denne plattformen er han og andre forskere ute av stand til å forstå hverandre faglig, og den tverrfaglige forskeren mister dermed også det eneste grunnlaget han har for å formidle faglig verdi og validitet fra forskningen sin. I henhold til dagens tradisjon for faglighet blir det dermed umulig å vurdere tverrfaglig forskning som foregår utenfor etablerte fagplattformer, og tverrfaglig forskning som foregår på spesialplattformer *uten* fagfeller, står dermed i fare for å bli oppfattet som "ufaglig".

Disiplinorientert eller "tverr faglighet"?

Etter mitt syn innebærer tverrfaglighet å bryte med etablerte fagretninger. Tverrfaglighet innebærer ikke bare nye utfordringer og å forske på andre måter (i betydningen "inter-faglig"); tverrfaglighet innebærer også ofte å gå *på tvers av* og *mot* etablerte disipliners orientering. Tverrfaglig er å være "tverr faglig" og kjerringa mot fagstrømmen, og det betyr ofte å *ikke* forske i henhold til etablerte disipliners interne faglige standarder og å *ikke* interessere seg for etablerte disiplinorienterte problemstillinger.

En slik "tverr faglig" orientering står i et motsetningsforhold til den disiplinorienterte forståelsen av faglighet og er derfor også problematisk å forene med denne. Dersom disse motstridende orienteringene forenes, vil forskeren måtte løse *både* utfordringene

i forhold til de interne, faglige standardene og de tverrfaglige utfordringene som går på tvers av disse. En slik tilnærming til tverrfaglighet kan dermed etablere en slags djevelsk "Ole Brum"-tradisjon som spør "Kan du ikke orientere deg både *med* og *mot* disiplinene samtidig?" og forlanger "Ja takk, begge deler!".

Dette motsetningsforholdet mellom tverr og disiplinorientert faglighet står sentralt i min forståelse av tverrfaglighet. Jeg vil derfor først ta utgangspunkt i en tverrfaglig tilnærming som forsøker å løse denne konflikten og samtidig ivareta disiplinorienterte tradisjoner. Denne tilnærmingen gir dessverre ikke et godt grunnlag for å forstå de tverrfaglige problemene i SjangDAI-prosjektet. Deretter vil jeg ta for meg en tverrfaglig tilnærming som går lengre i sin tverre faglighet og som langt mindre lytter til disiplinorienterte faglige standarder. Denne tilnærmingen stemmer langt bedre med våre erfaringer.

1.3 Tverrfaglig eller disiplinorientert? Ja takk, begge deler!

Det er flere beslektede tilnærminger som forsøker å forske både disiplinorientert og på tvers av faggrensene. Slike "i-pose-og-sekk"-tilnærminger kan hver på sin måte ha stor verdi (jf. Sørensen 2002: 14), men det kan være verdt å undersøke hvordan disse tilnærmingene faktisk går fram for å forene tverr og disiplinorientert faglighet. For å gjøre dette må vi granske hva som legges i begrepet "tverrfaglig", og hva som menes med å kombinere tverrfaglig og disiplinorientert forskning i et prosjekt.

Mye av debatten om tverrfaglighet ved NTNU tar utgangspunkt i slike "pose-og-sekk"-tilnærminger. Underforstått i denne debatten er ønsket om å skape nye tverrfaglige muligheter og samtidig beholde etablerte disiplinorienterte forutsetninger. For å få i stand denne kombinasjonen mellom tverr og disiplinorientert faglighet, assosieres gruppeaktiviteter fortrinnsvis med tverrfaglighet og enkeltforskerens aktiviteter med disiplinorientert faglighet. "Den tverrfaglige forskningen ved NTNU og andre steder baserer seg mer og mer på kollektive arbeidsformer, der forskjellige fagspesialister forsøker å arbeide sammen i team" (Sørensen 2002: 11). Gruppearbeid er også fokus for faget "Ekspert i Team" som er et tverrfaglig flaggskip ved NTNU (EiT 2003). Tverrfaglighet er et spørsmål om samarbeid mellom disiplinorienterte forskere, og et forskningsprosjekt kan dermed omtales som både disiplinorientert og tverrfaglig fordi det inkluderer både tverrfaglig gruppearbeid og enkeltforskeres disiplinorienterte arbeid.

Denne tilnærmingen krever at prosjektbegrepet utvides, og "et prosjekt" kan her omfatte alt fra løse diskusjoner til ferdig publiserte artikler. Men mange av disse aktivitetene trenger bare å være løst knyttet til hverandre, og tverrfaglige og disiplinorienterte aktiviteter skilles fra hverandre i forskjellige faser eller nivåer av "prosjektet". En vid prosjektbetegnelse kan dermed gi inntrykk av å ramme inn en

felles prosess, mens det i praksis rammer inn flere adskilte prosesser. Dette gir også inntrykk av at det foregår en syntese mellom faglige perspektiver fra flere forskjellige fagretninger, mens de forskjellige perspektivene i praksis forblir mer eller mindre upåvirket av hverandre.

I denne drøftingen vil jeg prøve å synliggjøre nivå- og tidsforskjellene i tverrfaglig samarbeid mellom disiplinorienterte forskere. Jeg vil her være restriktiv med hva jeg beskriver som tverrfaglig, og jeg vil også bare bruke begrepet ”prosjekt” om ”forskningsprosessen og arbeidet med problemløsning og kunnskapsutvikling fram mot og gjennom publisering”. I framtidens universitetssystem kan disse prosjektene bli de eneste forskningsaktivitetene som er verdt forskningsmidler (jf. UiO 2003: 4). Dette håper jeg vil tydeliggjøre min egen posisjon til tverrfaglighet og synliggjøre skiller jeg mener blir oversett. For meg blir derfor den sentrale problemstillingen *om det tverrfaglige samarbeidet endrer forutsetningene for og bruken av faglige perspektiv i et enkeltprosjekt?*

Tverrfaglig diskusjon, men disiplinorientert faglig arbeid

En form for tverrfaglig, disiplinorientert samarbeid kan beskrives som et ”diskusjonsfellesskap” (Sørensen 2002: 14). I et diskusjonsfellesskap kan ”ideer og synspunkter utveksles relativt uforpliktende” (ibid.), og deltakerne er fremdeles spesialister med hver sine disiplinorienterte problemer og agendaer.

Diskusjonsfellesskap er ganske åpne, generelle plattformer, der spesialistene kan komme med kryssfaglige innspill. Slike innspill er friske pust som gir spesialistene nye innfallsvinkler de kan ta med tilbake til sine respektive fagområder og bruke til å utdype eller utvide sin forståelse av disiplinorienterte problemstillinger.

Diskusjonsfellesskapet er altså ikke selv opptatt av å takle kompliserte problemer. Slike oppgaver krever spesialisering, og denne samarbeidsformen overlater derfor problemløsingen til den enkelte forsker å løse innenfor etablerte, spesialiserte fagretninger. Diskusjonsfellesskapet blir dermed en kreativ, uforpliktende og generell plattform som er ment å støtte oppunder disiplinorientert forskningsarbeid på spesialplattformene.

For meg fortøner diskusjonsfellesskapet seg lett som en arena der spesialistene får et slags friminutt fra sitt egentlige arbeid. Tilbake ved arbeidspulten forventes forskeren å igjen bruke faglige ideer og perspektiv i henhold til de interne faglige standardene i sitt fagområde. I sluttspurten fram mot publisering er dermed kryssfaglige innspill ikke lenger uforpliktende, men derimot forpliktet til å forholde seg til den etablerte fagretningen og dens interne, faglige standarder.

Diskusjonene kan derfor ha tverrfaglig verdi, men ettersom disse diskusjonene ikke er en direkte del av forskningsarbeidet, overstyrer ikke diskusjonsfellesskapet de disiplinorienterte føringene som gjelder for det enkelte forskningsprosjektet. Arbeidet i enkeltprosjektet bør fremdeles skje i henhold til interne faglige standarder innenfor etablerte fagområder, og diskusjonsfellesskapet setter dermed ikke forskjellige fagretninger opp mot hverandre i prosjektet. Det oppstår dermed et skille mellom det tverrfaglige motsetningsforholdet som eksisterer i diskusjonsgruppa og den disiplinorienterte forskningen som deltakerne gjør i egne prosjekt.

Dersom deltakerne i et prosjekt primært bruker faglige perspektiv i henhold til interne faglige standarder innenfor et etablert fagområde, er dette prosjektet disiplinorientert. Et slikt prosjekt kan heller ikke automatisk påberope seg tverrfaglig status, selv om det skulle være inspirert av faglige innspill utenfra. Dersom slik status overføres fra diskusjonsfellesskapet til disiplinorienterte enkeltprosjekt, står vi i fare for å overse at også enkeltprosjekt kan ha en tverrfaglig profil. Dette kan undergrave slike prosjekts problemer knyttet til tverrfaglig konstruksjon av spesialplattformer og påfølgende kommunikasjon. En slik sammenblanding opplever jeg som en reell trussel mot vårt tverrfaglige prosjekt.

Flerfaglig samarbeid

I diskusjoner om tverrfaglighet er det også vanlig å skille mellom *flerfaglig* og *tverrfaglig* forskning. Både flerfaglig og tverrfaglig forskning bruker perspektiv fra flere forskjellige fagretninger, men flerfaglig forskning er mer disiplinorientert enn tverrfaglig forskning.

For å synliggjøre hvordan flerfaglig samarbeid skiller mellom tverrfaglige og disiplinorienterte aktiviteter, har jeg valgt å dele flerfaglig samarbeid inn i to retninger. Dette skillet er ikke en beskrivelse av eksisterende flerfaglig forskning, men et retorisk grep for å vise hvordan nivå- og tidsforskjeller holder tverrfaglige og disiplinorienterte aktiviteter fra hverandre.

Flerfaglig samarbeid som overordnet prosjektramme

En form for flerfaglig samarbeid ”innebærer en mindre tett sammenkobling av bidrag fra flere fag rettet inn mot å belyse et felles overgripende problem” (Mathisen 1997: 29). I motsetning til diskusjonsfellesskap som bare utveksler faglige perspektiv, er det her tale om arbeid innen en felles ramme og samarbeid mot et felles overgripende problem. Men det overgripende problemet er gjerne definert slik at det blir få og små koblinger mellom de fagspesifikke bidragene. Flerfaglig samarbeid blir dermed en slags overordnet ramme der en gruppe forskere arbeider mot et felles mål, men utfører hver sine disiplinorienterte deloppgaver.

Dersom f. eks. det overordnede problemet er en skitten elefant og målet er en ren elefant, kan en flerfaglig forskergruppe dele elefanten inn i soner og vaske problemet fra hver sin kant. En slik overordnet samarbeidsramme utfordrer i liten grad den grunnleggende disiplinorienteringen hos den enkelte forskeren: hver kan vaske sin elefantsone på sin egen, disiplinorienterte måte. Gitt en slik overordnet oppdeling av problemet, vil samarbeidet raskt utvikle seg til en gruppe av løst sammensatte disiplinorienterte delprosjekt².

Samarbeid på tvers av faggrensene behøver heller ikke å bety at et flerfaglig samarbeid bryter med disiplinorienterte forutsetninger. En elektroingeniør kan f. eks. være vant til å levere faglige forståelsesmodeller til en dataingeniør, og en anvendt språkviter kan f. eks. ha lang erfaring med å ta i bruk sosiologiske teorier. Det kan derfor være helt i tråd med faglige tradisjoner at fagspesialisten bruker resultater fra visse andre fagretninger innenfor en overordnet ramme.

Dersom koblingen mellom delprosjektene er ”mindre tett” og følger etablerte mønster for faglig utveksling, er det i det enkelte delprosjektet få grunner til å gå på tvers av eller mot interne faglige standarder. I eksempelet med elefanten har hver enkelt forsker bare ansvar for å vaske sin egen sone. Forskerne har derfor også liten egeninteresse i å hjelpe og lære av hverandre. Summen av frittstående prosjekt kan derfor i en overordnet ramme forstås som flerfaglig, mens hvert enkelt delprosjekt forblir disiplinorientert.

Flerfaglig samarbeid som forstadium for disiplinorientert arbeid

En annen måte å forstå flerfaglig samarbeid på, er som et tverrfaglig prosjekt som etter hvert spaltes opp i flere mindre disiplinorienterte prosjekter. I motsetning til flerfaglig samarbeid som overordnet prosjektramme, er det her tale om et skille mellom tverrfaglige og disiplinorienterte aktiviteter i forskjellige faser av prosjektet.

I startfasen av prosjektet oppfordres den enkelte forskeren til å forholde seg til den tverrfaglige gruppa og bruke sine faglige perspektiv slik at de passer sammen med resten av gruppas. I eksempelet med den skitne elefanten, kan gruppa f. eks. sammen gå rundt elefanten og bygge flerfaglig konsensus om at elefanten bør spyles, ikke skrubbes. I slike situasjoner må deltagerne i langt større grad bli enige om hva det felles problemet innebærer og hvordan gruppa samlet skal håndtere det. Som grunnlag for å skape større faglig enighet, trenger gruppa derfor først å etablere en begynnende plattform av gjensidig, samlende faglig forståelse. Dette krever at gruppa arbeider med å forstå hverandres faglige perspektiv og å integrere hverandres faglig forståelse.

² Elefantmetaforen er gitt meg av Ellen Andenæs under personlig veiledning sommeren 2003.

Denne integreringen behøver ikke å være disiplinorientert og kan gå på tvers av og gjerne mot etablerte fagretningers standarder.

Men i sluttfasen av prosjektet inverteres denne forskningsplattformen og legger opp til et åpnere tverrfaglig samarbeid basert på spredt, disiplinorientert faglig arbeid. Forskningsresultatene er ”fagspesifikke og publiserer[s] hver for seg” (Sørensen 2002: 14), og dersom fagspesifikke standarder kommer i konflikt med den tverrfaglige bruken av perspektivene, har ikke gruppa noe felles ansvar for syntetisering. Tilbake til eksempelet med elefanten, kan forskerne etter å ha etablert en felles konsensus igjen dele elefanten inn i soner. Gruppa har etablert noen felles retningslinjer, men deretter overlates den enkelte forskeren til seg og sitt delproblem.

Deltagerne arbeider dermed disiplinorientert fram mot og gjennom publisering. Et eventuelt tverrfaglig arbeid med å utvikle gjensidig forståelse og relevans innad i gruppa, skyves gradvis ut på sidelinja grunnet publiseringens betydning. Deltagerne forventes å forlate den tverrfaglige plattformen de har bygget opp, og returnere til de tradisjonelle, fagspesifikke plattformene med sine flerfaglige resultater. Gradvis opptrer dermed denne samarbeidsformen mindre forpliktende ovenfor det tverrfaglige samarbeidet og mer forpliktende ovenfor disiplinene.

Dette plattformbyttet er tvetydig med hensyn til hvilke rammer som gjelder for faglig arbeid. Skal forskeren forholde seg til den tverrfaglige gruppa og problemene de sammen prøver å løse, eller skal forskeren overføre problemene fra denne gruppa ”tilbake” sitt eget, spesialiserte fagområde og løse dem der? Flerfaglig samarbeid legger primært opp til at enkeltforskeren skal overføre problemer og resultater til sitt eget fagområde. Publisering i etablerte fagfora er et så viktig mål at faglige krav og standarder for enkeltprosjektet forblir disiplinorienterte. Den tverrfaglige gruppas behov og ståsted som har prioritet i startfasen, blir mot slutten av prosjektet dermed underordnet disiplinorienteringen.

Problemer med en flerfaglig fase i et disiplinorientert prosjekt

Et sentralt problem med en slik flerfaglig snuoperasjon er at den forutsetter at faglige problem, resultat og perspektiv kan overføres fra en faglig plattform til en annen. Men dette er ikke nødvendigvis tilfelle. Når den faglige rammen rundt slike perspektiv endrer seg, kan dette også føre til at perspektivene endrer karakter. Et problem vil f. eks. fortone seg annerledes i forhold til hvilken metode forskeren velger å bruke for å løse det, og metoden belyses i sin tur forskjellig avhengig av bl. a. forskerens teoretiske bakgrunn. Den flerfaglige plattformen i startfasen av prosjektet kan dermed utgjøre en usynlig bakgrunn som er en del av de faglige perspektivene, og for å forstå f. eks. hva den flerfaglige gruppa mente med ”å spyle elefanten”, må man derfor se den skitne elefanten slik den flerfaglige gruppa gjorde det.

Faglige perspektiv, problem og resultat blir dermed avhengig av hverandre. Hvert eneste perspektiv har en bakgrunn av andre perspektiv og forutsetninger som de trenger for sitt underhold. Faglige perspektiv danner en bakgrunn for hverandre, og perspektivene knytter seg dermed sammen i lange kjeder til de til slutt blir til et slags flokete garnnøste. Og fordi faglige perspektiv, problem og resultat er avhengige av hverandre, kan de hver for seg derfor heller ikke trekkes ut av sin faglige sammenheng og veves inn i en annen uten problemer.

Forskere som ønsker å overføre et faglig perspektiv *fra* en flerfaglig plattform, må derfor først undersøke den flerfaglige bakgrunnen dette perspektivet befinner seg i. En slik undersøkelse vil bringe fram andre forutsetninger og perspektiv som i sin tur også har en egen bakgrunn og egne forutsetninger. Men perspektivene som skal overføres, har ikke bare en etablert bakgrunn forskerne må ta hensyn til. Forskere som ønsker å overføre faglige perspektiv *til* en annen spesialplattform, må også ta hensyn til den *nye* faglige bakgrunnen. Forskerne må derfor også undersøke hvordan perspektivene passer sammen med de interne, faglige standardene på denne nye plattformen. Denne undersøkelsen vil gi et inntrykk av hvor mange andre faglige forutsetninger det faglige perspektivet støtter seg på og derfor må overføres parallelt.

Hensynet til bakgrunnen som perspektivet skal overføres *til*, legger i praksis sterke begrensninger på hvor fremmed et tverrfaglige innslag kan være. Når forskeren henter inn og bruker av fremmedfaglige perspektiv innenfor sitt spesialfelt, vil disse også gjøre krav på at dette spesialfeltet rommer andre faglige perspektiv som disse fremmedfaglige perspektivene forutsetter. Dersom disse forutsetningene også er fremmede innen den nye faglige bakgrunnen, må disse også overføres. Blir ikke slike usynlige, fremmede forutsetninger synliggjort, vil problemene og resultatene kunne virke svært fremmede og faglig ugyldige i henhold til de interne faglige standardene innen fagretningen.

Forskere som ønsker å bruke faglige perspektiver mot en ny, faglig bakgrunn, må derfor nøye undersøke om denne bakgrunnen har tilsvarende forutsetninger som den gamle, faglige bakgrunnen som perspektivene pleide å figurere mot. Er forskerne gjerrige med hvilke forutsetninger de synliggjør sammen med det fremmedfaglige perspektivet, risikerer de at problemene ikke belyses fra rett side og at resultatene virker som om de er tatt rett ut av løse lufta. Men dersom overføringen av fremmede perspektiv blir for kraftig, vil andelen ukjente forutsetninger som må presenteres bli for stor. Forskeren må derfor begrense utvalget av faglige problem og resultat til de som kan trekkes opp fra den flerfaglige plattformen og overføres uten alt for mange nisser på lasset.

Eventuelle konflikter mellom nye tverrfaglige innspill og etablerte, interne faglige standarder, må derfor balanseres slik at de nye innspillene kan veves sammen med de etablerte standardene. En slik balansegang vil måtte ta sterke hensyn til disiplinorienterte, faglige standarder, og i slutfasen når prosjektet går mot publisering får disiplinorienteringen det siste ordet. Tverrfaglige perspektiv som ikke passer sammen med disiplinenes forutsetninger eller som ikke kan innordne seg etter disse, må forbli tause. I praksis ender derfor tverrfaglige innspill fort opp som ”det lille ekstra” og pynt i et ellers disiplinorientert forskningsprosjekt.

Hva med flerfaglig utvikling?

Et annet og kanskje enda mer underfundig problem med flerfaglig faseinndeling, er at denne inndelingen gir inntrykk av at enkeltforskeren ikke utvikler seg faglig i den flerfaglige samarbeidsgruppa. Forskeren forventes å gå i gang med samarbeidet med sin spesialfaglige bakgrunn og sine spesialfaglige perspektiv. I denne flerfaglige omgivelsen skal så disse perspektivene brukes sammen med andre spesialister fra andre fagområder. I dette samarbeidet lærer fagfolkene av hverandre og utvikler sammen ny forståelse basert på felles ressurser. Deretter returnerer fagspesialistene til sitt spesialfelt med sin faglige bakgrunn og sine perspektiv intakt. Spesialistene og deres spesialfaglige perspektiv og bakgrunn forblir uendret både på vei inn i, underveis og på vei ut av det flerfaglige samarbeidet.

Dette er ikke uproblematisk. Jeg har allerede påpekt at problem og resultater utviklet på en flerfaglig bakgrunn, ikke uten videre kan transporteres fra plattform til plattform. Men det er heller ikke sikkert at forskeren selv og hans spesialkompetanse ubekymret kan hoppe fra plattform til plattform. Hva skjer når forskeren faktisk lærer noe underveis, noe han ikke kan sette ord på og kanskje selv ikke er klar over? Hva om de andre spesialistene smitter ham med noe av deres egen tause kunnen? Når forskerne lærer av hverandre og tilpasser sine forutsetninger for og bruk av faglige perspektiv til hverandre, er det slett ikke sikkert at de er bevisst alle endringene de gjør og alle de nye faglige perspektivene de tilegner seg.

Når verken publisering eller syntetisering på tvers av fagretninger faller inn under det flerfaglige samarbeidets ansvarsområde, blir ingen av disse problemene behandlet der. Videre forutsetter den disiplinorienterte forståelsen av faglighet at faglig utvikling skjer i henhold til interne faglige standarder, og dette gjelder ikke bare for enkeltprosjekter og fagområdene som helhet, dette gjelder også for personlig faglig utvikling. Det betyr at verken det flerfaglige samarbeidet eller de tradisjonelle rammene for forskning tar hensyn til den personlige, faglige utviklingen som skjer i henhold til både interne og eksterne, faglige standarder.

Mathisen (2002) peker på at manglende anerkjennelse for flerfaglig utvikling er et stort personlig problem for flerfaglige og tverrfaglige forskere. Innenfor spesialistregimet faller deres form for faglig utvikling utenfor etablerte faglige og institusjonelle rammer. Ikke bare utsettes faglige problem og resultater for et ”disiplinært trykk” (op. cit.: 33f), men fagpersoner må også pent falle inn i rekkene og trampe i takt. Fagfolk som utvikler seg flerfaglig, kan derfor i liten grad bidra til eller vurderes etter ”disiplinenes kriterier” (ibid.) og havner dermed på sidelinja i akademia.

1.4 Tverrfaglig eller disiplinorientert? ”tverr faglig”, takk!

Tverrfaglig samarbeid

Sørensen (2002: 11) beskriver tverrfaglig samarbeid som ”en arbeidsform der en tar sikte på å frambringe resultater som er felles, og som ikke hører hjemme i ett fagfelt”. Slikt samarbeid har gjerne ”et felles ansvar for å sy sammen deløsninger til en helhet” eller å produsere ”felles publikasjoner” (ibid.). I motsetning til flerfaglig samarbeid, blir dermed det tverrfaglige samarbeidet mer forpliktende ovenfor den enkelte forskeren, og tverrfaglig samarbeid etablerer en overordnet ramme som påvirker forskernes arbeid fra flere fagretninger samtidig gjennom hele prosjektet.

Det er flere årsaker til at forskere velger en tverrfaglig forskningsstrategi. Noen er ”problemorienterte”, mens andre er ”kunnskapskritiske”, dvs. kritiske til disiplinenes spesialisering og manglende helhetsperspektiv (Dahl 1997: 9). I denne teksten diskuterer jeg utelukkende tverrfaglig forskning som er problemdrevet og som følger tverrfaglige problem fra start til mål. Eventuelle konsekvenser for disiplinorientering og spesialisering diskuterer jeg bare som en bivirkning av dette.

Tverrfaglig arbeid krever utadvendthet og flerfaglig læring

Når faglige perspektiv fra forskjellige fagretninger brukes samtidig, danner de en bakgrunn for hverandre. En slik flerfaglig bakgrunn kan også forstås som nye faglige forutsetninger og blir en nødvendig del av helhetsbildet i et tverrfaglig prosjekt. Tverrfaglige forskere trenger derfor å sette seg inn i forutsetninger fra flere fag samtidig, uavhengig av om disse kommer fra hans opprinnelige spesialfelt eller ikke.

Den tverrfaglige virkeligheten driver dermed forskere til å tilegne seg faglig innsikt i flere fagretninger. Åpenhet, vilje og ferdigheter i å tilegne seg slik fremmedfaglig innsikt blir dermed et kjennetegn ved tverrfaglig samarbeid og et krav til den tverrfaglige forskeren (jf. ”polyvalens” (Sørensen 2002: 17f)). Den enkelte tverrfaglige forskeren kan ikke forbli en spesialist *bare* innenfor sitt eget fagfelt, og spesialistene går dermed fra å være faglig innadvendt (jf. ”fagidiot”) til faglig

utadvendt. Tverrfaglig utvikling blir dermed sideordnet disiplinorientert utvikling, både for enkeltforskeren, samarbeidsgruppa og forskningsmiljøet rundt.

Det tar tid å være utadvendt og lære fremmede fagretningers indre logikk og sammenheng. I tverrfaglig samarbeid trenger derfor deltakerne å ”bruke tid på hverandre og utvikle gjensidig forståelse og respekt” (Evensen 2002a: 134). Men slik tidsbruk er ikke allment anerkjent, og ”det omfattende tidkrevende arbeidet som ligger bak tverrfaglig forskning med hensyn på å skape gjensidig forståelse mellom forskere med ulik bakgrunn [underkjennes ofte]” (Mathisen 1997: 33). Jeg støtter derfor arbeidet med å få anerkjent utadvendthet og flerfaglig læring som nødvendig del av tverrfaglige prosjekt (jf. Mathisen 1997, Sørensen 1997 og 2002, Evensen 2002a).

Tverrfaglig arbeid krever tilpasning av faglige perspektiv

Faglig utadvendthet og flerfaglig læring er ikke tilstrekkelig for vellykket tverrfaglig samarbeid. ”Spesialister har et blikk og et språk som er særegent” (Sørensen 2002: 15), og det kan derfor være vanskelig å samordne perspektiv fra flere fag uten å skape ”en babelsk forvirring” (Dahl 1997: 13). Faglige perspektiv og verktøy som brukes samtidig, kan også produsere motstridende, usammenhengende og forvirrende resultater. Slik tverrfaglig forvirring påvirker samarbeidet og kommunikasjonen forskjellige spesialister imellom (Sørensen 2002, Evensen 2002a). Men slike tverrfaglige forskjeller kan også forvirre enkeltforskeren og få forskjellige deler av hans aktiviteter til å motarbeide hverandre. Håndtering av tverrfaglig gruppeforvirring må i tillegg basere seg på at enkeltforskere først greier å forsone de forskjellige tverrfaglige perspektivene. Dette er dermed like mye et problem for enkeltforskeren som for samarbeidsgruppa.

For å håndtere tverrfaglige forskjeller, må den enkelte forskeren derfor forsone og syntetisere fremmedfaglige perspektiv. Slik syntetisering krever at forskerne lærer perspektiv fra flere fag, men det krever også at spesialistene er villige til å tilpasse og gjennomføre endringer av de enkelte perspektivene. Det er meningsløst å snakke om samtidig bruk av faglige perspektiv uten å tilpasse disse til hverandre, og det er også meningsløst å snakke om tilpasning av faglige perspektiv uten å være villig til å endre de involverte perspektivene.

Etablering av en historieløs, men spesialisert faglig plattform

Tilpasning av perspektiv på tvers av faggrensene har mange likhetstrekk med tilpasning av faglige perspektiv innenfor samme fagretning. Men tverrfaglig tilpasning skiller seg også fra tilpasning innenfor samme fagområde i ett sentralt henseende.

Perspektiv fra samme fagretning har ofte en felles forhistorie der de har blitt sammenholdt og syntetisert. Perspektiv fra samme fagretning er *historisk beslektede* gjennom mange felles koblinger og forutsetninger, og dette gir forskerne tidligere prosjekter å bruke som eksempler til etterfølgelse. Mange disiplinorienterte prosjekt kan derfor forutsette at forskjellige perspektiv passer sammen.

Perspektiv fra forskjellige fagområder mangler derimot ofte en slik felles forhistorie og har derfor få felles forutsetninger og mange særegne forutsetninger. Når fremmede perspektiv skal syntetiseres med hverandre, har de derfor *både* færre felles, kjente forutsetninger *og* flere særegne forutsetninger enn disiplinorienterte, historisk beslektede perspektiv. Et tverrfaglig prosjekt bør derfor forutsette at dets forskjellige perspektiv ikke er kompatible.

Når perspektiv med få felles forutsetninger og mange særegne forutsetninger skal syntetiseres, utsettes det enkelte perspektivet for sterke tilpasningskrav. Når avstanden mellom de faglige perspektivene blir stor nok³, kan disse kravene oppfattes som en konflikt. Det sterke ytre presset fører videre til et sterkt indre press på disse perspektivene om å utvikle seg. Fremmedfaglige perspektiv må strekkes lenger for å komme hverandre i møte, og under et slikt press beveger gjerne det enkelte perspektivet seg langt vekk fra sitt utgangspunkt. Det store spennet mellom fremmedfaglige perspektiv fører dermed til at de enkelte perspektivene må strekkes så langt at de kan miste kontakten med sine opprinnelige interne faglige standarder.

Fremmede perspektiv tilpasses heller ikke bare hverandre, men blir også mer forståelige for de forskjellige spesialistene i gruppa. Utviklingen er med på å tilgjengeliggjøre fremmede perspektiv innad i det tverrfaglige prosjektet. "Vidstrakt" utvikling av enkelte, faglige perspektiv i et tverrfaglig prosjekt er altså drevet fram av behovet for å syntetisere fremmedfaglige perspektiv med hverandre. Men det som oppfattes som en naturlig tilpasning eller videreutvikling innad i den tverrfaglige gruppa, kan oppfattes som fremmed og spesialisering sett fra perspektivenes opprinnelige spesialfelt. Den tverrfaglige tilpasningen gjør dermed perspektivene uforståelige og utilgjengelige for spesialister fra de opprinnelige fagområdene. Prosjektet mister dermed fotfestet i spesialistenes opprinnelige fagfelt, og utvikling som er drevet fram av problemorientering og utadvendt samarbeidsvilje, oppfattes lett som forsøk på å skjerme tverrfaglige prosjekt fra ytre innsikt og evaluering (jf. disiplinorientert spesialisering s. 9). Tverrfaglig utviklingen tilgjengeliggjør internt, men fremmedgjør eksternt.

³ Dersom vi skulle ønske å kvantifisere slike kvalitative koblinger, kunne vi f. eks. målt antall referanser som måtte til for å finne første felles vitenskapelige refererte eller refererende kobling.

Jo mer tverrfaglige perspektiv syntetiseres, desto mer vil dermed det tverrfaglige prosjektet etablere en egen spesialplattform av egne spesialperspektiv som er fremmed for andre fagfelt. Rammen rundt prosjektet vil utvikle seg fra å være en flerfaglig sum av perspektiv til å bli en tverrfaglig hybrid der perspektivene er farget fra flere hold. Den faglige bakgrunnen i et tverrfaglig prosjekt blir dermed ikke summen av de faglige perspektivene som tilpasses på tvers av flere fag, men *selve tilpasningen* av disse. Bakgrunnen blir en bro av nye perspektiv som strekker seg ut fra en faglig spesialplattform til en annen. Den tverrfaglige virksomheten og dets resultater vil foregå oppe på broen og er forskjellig fra selve broen.

Ettersom disse perspektivene er spesialistene syn på ting og tang, så vil disse endringene også påvirke spesialistenes identitet. Som annen faglig utvikling, vil mye av denne tverrfaglige, personlige utviklingen være vanskelig å oppdage og sette ord på. Taus kunnen blir dermed både et problem for tverrfaglige forskere både når de går inn i ett fremmed, tverrfaglig prosjekt og når de senere går ut av dette prosjektet igjen.

En ”tverr”, historieløs og spesialisert faglig plattform

Et tverrfaglig prosjekt kan dermed etablere en bakgrunn som virker både historieløs og spesialisert på samme tid. Hver for seg kan riktignok de faglige perspektivene som danner bakgrunnen i prosjektet ha en spesialistkompetanse og forhistorie, men sammensetningen av disse og den tverrfaglige bakgrunnen som helhet, blir ”banebrytende” og mangler tidligere forskningsprosjekt med lignende orientering som bakteppe. Tverrfaglige prosjekt får dermed en bakgrunn som helhetlig er historieløs, men som er satt sammen av spesialiserte historierike delperspektiv.

Den historieløse, helhetlige bakgrunnen blir både en mulighet og en trussel for tverrfaglige prosjekt. Å arbeide på et faghistorisk nytt område med spesialiserte faglige perspektiv, gir muligheter til å utvikle *mye* ny kunnskap *fort*. Det blir som å pløye gjennom et ukjent landskap med velkjente redskap. Likevel kan denne historieløsheten bli en veldig belastning. Dersom tverrfaglige prosjekt forventes å ha en etablert, helhetlig forhistorie som er vanlig i prosjekt med en mer ensrettet faglig orientering, så vil prosjektet selv måtte etablere denne forhistorien. Dette blir synlig når tverrfaglige prosjekt forventes å gå grundig til verks i f. eks. sine metodiske, teoretiske eller historiske drøftinger når disse sidene ved bakgrunnen oppfattes som ny og ukjent av omverden. Dette kapitlet er et eksempel på dette.

Å være ”tverr faglig” er dermed ikke bare å være utadvendt og lære faglige perspektiv fra flere fagfelt parallelt. Det ”tverr faglige” innebærer også å tilpasse disse perspektivene i forhold til hverandre og på den måten utvikle en særegen faglig bakgrunn. Denne bakgrunnen får en spesiell karakter fordi den som helhet mangler tidligere forskningsprosjekt å sammenligne seg med, mens enkeltelementer i denne

helheten derimot ofte har en lang, spesialisert forhistorie. Tverrfaglige prosjekt risikerer dermed å etablere en helt særegen form for spesialistkompetanse og en helt ny spesialisert plattform å arbeide på.

Nye tverrfaglige plattformers store kommunikasjonsproblem

Som beskrevet tidligere, er forskning grunnleggende kollektivt, og andre må få innsikt i forskningsprosessene slik at kunnskap skapt med bruk av faglige perspektiv blir delt. Forskerkollektivet er derfor ikke direkte interessert i om enkeltforskeren finner svar på sine spørsmål, men direkte interessert i de spørsmål og svar han greier å dele med dem.

Når tverrfaglig tilpasning etablerer en ny og spesialisert fagplattform, mangler etablerte fagfelt den nødvendige spesialistkompetansen for å forstå spørsmål og svar som denne nye plattformen underbygger. Forskere på en slik ny og spesialisert fagplattform kan heller ikke forvente at det skal finnes andre forskere som er inneforstått med dens spesialiserte, faglig sammensatte oppbygning. Dermed blir plattformen omgitt av utenforstående.

Fraværet av inneforståtte spesialistkolleger er et vesentlig brudd med disiplinorientert tradisjon. Disiplinorientert forskning baserer seg på at det finnes disipliner som forstår den faglige bakgrunnen i prosjektet og som deler interessen for prosjektets spørsmål og svar. Disiplinorientert, faglig kommunikasjon foregår *på* fagfeltenes plattformer, og alle deltagerne forutsettes å ha den nødvendige faglig kompetansen til å delta. Disiplinorientert kommunikasjon forutsetter altså at faglig bakgrunn og spesialisering i prosjektet og disiplinen til en stor del overlapper hverandre, men denne forutsetningen er ikke tilstede i et tverrfaglig prosjekt. I et tverrfaglig prosjekt er det derimot utviklet en egen spesialisert fagretning, og faglige metoder og perspektiv er brukt mot en mer eller mindre unik faglig bakgrunn. Desto mer ”banebrytende” tilpasningen av spesialiserte perspektiv er, desto mindre vil bruken av disse være forståelig for etablerte, omkringliggende disipliner.

Problemet for tverrfaglig forskere er at de - til tross for dette dilemma - ikke har noen andre mer nærliggende fagplattformer og kommunikasjonsarenaer å dele sin kunnskap på. Så lenge forutsetningen om at forskning er kollektiv og at kunnskap skal deles blir stående, må tverrfaglige forskere snu seg mot disiplinene likevel. Dette skaper en mismatch, og ”komiteene som foretar faglige vurderinger [mangler derfor ofte] kompetanse til å vurdere det vitenskapelige arbeidet som et tverrfaglig arbeid, [og de vurderer derfor arbeidet] ut fra tradisjonelle disiplinære kriterier” (Mathisen 1997: 33). Dette kan skape store kommunikasjonsproblem.⁴

⁴ Dette har ført til at vår bruk av faglig spesialiserte metoder, teorier, problemer og løsninger er uinteressante som faglige spørsmål for utenforstående. Våre tverrfaglige resultater og svar på slike

Nystrand del 1: Tverrfaglige skrivere, disiplinorienterte lesere

For å belyse slike tverrfaglige kommunikasjonsproblemer, vil jeg bruke sosio-interaksjonistisk kommunikasjonsteori (Nystrand 1986, 1989). Nystrand (1986) beskriver skriftlig kommunikasjon som en prosess der deltakerne "forhandler" om mening. I denne forhandlingsprosessen "leser leseren på skriverens premisser, og skriveren skriver på leserens premisser" (Rommetveit 1974: 63 sitert etter *ibid.*: 41). Leseren og skriveren må altså forsøke å tilpasse det de vet fra før og ønsker å oppnå gjennom teksten med det de tror den andre parten vet fra før og ønsker å oppnå gjennom teksten.

Både skriver og leser kommuniserer dermed ut fra den andres ståsted og må utvikle felles forståelse stegvis. Men i faglig kommunikasjon er det først og fremst publisering som gjelder, og i disse prosessene er det forskeren som skriver som er oppe til vurdering (Mathisen 1997: 31, jf. UiO 2003). Det er derfor langt viktigere for skriveren å få leseren med seg på veien enn motsatt, og leserens forkunnskaper og interesser veier tungt når skriveren velger utgangspunkt og fokus i teksten. I tillegg til dette er også kommunikative arenaer som tidsskrift, institutter, konferanser osv. tett knyttet til disiplinene, og en disiplin er på mange måter konstituert av slike arenaer (Dahl 1997: 11f). Disse arenaene har tradisjoner for hvilke faglige forkunnskaper og kunnskapsinteresser lesere forventes å ha (Mathisen 1997: 31), og skrivere som ønsker å dele sin kunnskap på slike disiplinerte arenaer, må derfor forholde seg til disses interesser og forkunnskap.

Men en tverrfaglig forsker og skriver kan ikke bare se bort ifra sine egne interesser, forkunnskap og bakgrunn. Forskerens (tverr)faglige spørsmål og svar er basert på denne bakgrunnen, og forskersamfunnet kan ikke forvente at disse spørsmålene og svarene skal beskrives på en plattform uten noen former for forkunnskaper og interesser. All kommunikasjon baserer seg på forkunnskaper og interesser (Nystrand 1986, 1989), og tverrfaglig spesialisering av spesialiserte aktiviteter kan ikke - i likhet med annen spesialisert forskning - beskrives og forklares med et hverdagsspråk. En tverrfaglig forsker kan heller ikke utelukke sider ved forskningsprosjektet sitt som baserer seg på tverrfaglige forkunnskaper og interesser. Dette vil skjerme sider ved forskningen basert på en vurdering av relevans som kan virke helt malplassert sett fra et anvendt eller kunnskapssøkende ståsted, og det er heller ikke sikkert at prosjektet etter en slik vurdering vil ha noen resultater igjen å presentere. Dette vil i sin tur føre

spørsmål basert på arbeidet med disse spørsmålene blir videre enda mer fremmede, lite relevante og uinteressante for utenforstående. Det er snakk om to forskjellige spesialplattformer eller "kunnskapssituasjoner" (Haavind 2002). Den ene kunnskapssituasjonen er basert på flere fag og er utviklet i starten av prosjektet og danner dermed uunnngåelig et grunnlag for senere forskning. Denne definerer skriverens premisser. Den andre kunnskapssituasjonen er den som finnes innenfor de forskjellige spesialfeltene og det er denne som definerer leserens premisser.

til mindre uttelling eller mangedoblet arbeidsbelastning for den enkelte tverrfaglige forskeren (Mathisen 1997).

En tverrfaglig forsker (skriver) kan heller ikke se bort fra disiplinorienterte spesialisters (leseres) interesser når han kommer med faglige bidrag til kollektivet. Disiplinenes tradisjoner definerer hvilke forkunnskaper skriveren kan ta utgangspunkt i og hvilken retning forskeren skal spesialisere seg i. Forskeren kan dermed ikke ta utgangspunkt i den faglige bakgrunnen eller interessene fra prosjektet når han skal dele dette med en disiplin dersom denne bakgrunnen og interessen ikke stemmer overens med disiplinens. Resultatet blir at forskeren verken kommer unna sitt eget prosjekts eller disiplinens forutsetninger og interesser. Dersom tverrfaglige forskere ikke kan anse at bakgrunnen for arbeidet i prosjektet er felles forståelig for disiplinen, må han derfor først redegjøre for denne faglige bakgrunnen før han kan presentere resten av prosjektet.

Disiplinorienterte prosjekt må også redegjøre for den faglige bakgrunnen arbeidet i prosjektet baserer seg på. Men ettersom disse prosjektenes og disiplinens bakgrunn gjerne overlapper på mange områder, så kan disiplinorienterte prosjekt derimot redegjøre for sin posisjon i forhold til en felles kjent bakgrunn gjennom bl. a. å referere til felles kjente tidligere forskningsprosjekt. Basert på en relativt stor andel av felles kjente faglige perspektiv i forhold til ukjente faglige perspektiv, blir denne jobben som oftest overkommelig.

Tverrfaglige forskere (skrivere) har derimot gjerne en relativt større del av spesialiserte fagperspektiv som er fremmede for lesere innenfor disiplinene. Ettersom referanser til slike perspektiv er ukjent for leserne, er det ikke tilstrekkelig å bare peke på til slike fremmede perspektiv. Slike perspektiv må da presenteres og forklares, men dersom disse perspektivene krever spesialkompetanse for å forstås, vil slike forklaringer bli svært omfattende. I tillegg til slik omfattende presentasjon av den store fremmedfaglige bakgrunnen i prosjektet, vil også den tverrfaglige forskeren måtte redegjøre for hvordan disse fremmedfaglige perspektivene har blitt utviklet for å tilpasses hverandre. Mens referanser i disiplinorientert kommunikasjon dermed ofte fungerer som gjenkjennelsesmekanismer, fungerer referanser i en tverrfaglig kommunikasjonssituasjon derimot ofte som potensielle overraskelser.

Dersom tverrfaglige prosjekt forventes å kommunisere og dele sine resultat på lik linje med disiplinorienterte prosjekt, vil de dermed bli gitt dobbel arbeidsbelastning. Først må de forklare utfyllende den delen av den faglige bakgrunnen som er fremmed for den disiplinen de henvender seg til. Dette kan være en umulig oppgave ettersom disse spesialperspektivene kan ta flere år å lære. Deretter kan de begynne på den egentlige oppgaven med å presentere sin forskning og sine resultater produsert på

denne bakgrunnen. Å gjøre seg forstått av utenforstående ikke-spesialister kan dermed bli et massivt hinder og en tvangstrøye for den tverrfaglige forskeren.

Nystrand del 2: tverrfaglig fokus i teksten

Nystrand (1990) går lenger i å problematisere konflikten mellom skriverens og leserens interesser og forkunnskaper. Når skriveren ønsker å fremme et poeng, kan han og leseren ha forskjellige forkunnskaper og interesser knyttet til poenget. Skriveren stilles da ovenfor et valg om hvor fokuset skal ligge i teksten. Dersom han ønsker å få leseren til å forstå poenget slik han selv ser det, må han utdype den forkunnskapen og interessen han assosierer med poenget for leseren. Han må forklare den faglige bakgrunnen for det faglige poenget eller perspektivet han ønsker å fremme i teksten. Men disse forkunnskapene og interessene som ligger bak det opprinnelige faglige poenget, er også faglige poeng som også er basert på faglig forkunnskap og kunnskapsinteresser. Dersom leseren heller ikke deler disse, må skriveren også forklare den faglige bakgrunnens faglige bakgrunn. Slik kan skriveren måtte grave etter og begrunne sine faglige perspektiv og poeng i det uendelige, helt til han og leseren til slutt finner kunnskapsinteresser og faglige forkunnskaper de kan enes om.

En slik kjede av begrunnelsens begrunnelse er en skrivers mareritt. Skriveoppgaven og arbeidsbyrden vil øke dramatisk. En slik iterativ løkke vil også flytte fokuset fra det opprinnelige poenget til bakgrunnens bakgrunn for dette. Fokuset i teksten flyttes, og den tverrfaglige bakgrunnen og dennes faglige opphav kommer i fokus. Den tverrfaglige forskeren presses til å redegjøre for sin spesialisering, og ikke for resultatene som er arbeidet fram med disse spesialressursene. Slikt press kan føre til at den tverrfaglige forskerens egentlige faglige perspektiv og interesse utvannes til det nesten er fjernet fra teksten.

Den tverrfaglige forskeren får i oppgave å løfte leseren frem til prosjektets faglige nivå, ikke bare å referere til en felles kjent bakgrunn og forhandle om mening basert på en slik bakgrunn som disiplinorienterte skrivere. Min argumentasjon her kan virke som en gjerrig forskerposisjon, og jeg ville vært enig i en slik kritikk, hadde oppgaven med å sette leseren inn i den spesialiserte bakgrunnen i prosjektet vært overkommelig. Men ofte er ikke denne oppgaven det. Ofte har den tverrfaglige spesialiseringen i prosjektet tatt mange år og baserer seg på flere års ekstra studier i en annen faglig retning. Disse årelange studiene ”smitter” den tverrfaglige forskeren med mye taus kunnen. Oppgaven med å løfte andre opp på et slikt spesialiseringsnivå er derfor en oppgave som i utgangspunktet burde forutsette flere års trening og samstudier og er ikke et spørsmål om f. eks. ett ekstra avsnitt eller delkapittel som kan forklare en ikke-spesialisert leser bakgrunnens bakgrunn.

Mitt poeng med denne dramatiseringen av tverrfaglige kommunikasjonsproblem, er å synliggjøre at tverrfaglig spesialisering kombinert med disiplinorienterte tradisjoner kan skape umulige utfordringer og skriveoppgaver. Underforståtte forutsetninger og taus kunnen er ikke noe som bare kan feies under teppet. Dette gjelder i aller høyeste grad også disiplinorientert forskning, og mange av problemene og skriveoppgavene framhevet her gjelder derfor for både tverrfaglig og disiplinorientert kommunikasjon. Men mitt poeng er at dagens tradisjoner som imøtekommer disse problemene i disiplinorienterte prosjekt, kan feie beina vekk under tverrfaglige prosjekt. Mathisen (1997), Sørensen (2002) og Evensen (2002a) framhever alle tverrfaglig kommunikasjon som et stort problem, men jeg mener at dette problemet er større, mer grunnleggende og treffer den enkelte tverrfaglige forskeren og hans prosjekt kraftigere enn disse gir inntrykk av. Problemene med taus kunnen, underforståtte faglige forutsetninger og spesialisering treffer tverrfaglig forskning med dobbel tyngde.

Oppsummering av det tverrfaglige kommunikasjonsproblem

Tverrfaglige prosjekt møter en stor utfordring når de skal dele og kommunisere sitt arbeid og sine resultater. Den tverrfaglige forskeren blir stilt ovenfor et valg: enten kan han akseptere kravet om å skrive på leserens premisser og redegjøre for den tverrfaglige bakgrunnen med et disiplinorientert fagmiljø, eller så kan han gi blaffen i dette og presentere resultater og bruk av faglige perspektiv på sine egne premisser.

Problemet med å skrive på leserens premisser er at dette vil medføre at forskerens *resultater* ikke ser dagens lys. Det kan alltid argumenteres for at forskere selv er ansvarlige for først å publisere og gjøre felles materialet som er nødvendig for senere å publisere sine resultater på dette grunnlaget. Men disse argumentene baserer seg på forutsetningene om at arbeidet med å etablere dette felles grunnlaget er overkommelig innenfor den tverrfaglige forskerens ressurser, og at en bred, felles forståelse er viktigere enn en spisset, individuell forståelse. Problemene og resultatene i dette prosjekt setter derimot spørsmålsteget ved begge disse forutsetningene. For det første viser erfaringene fra dette prosjektet at det ikke nødvendigvis er tilstrekkelig med både faglige og praktiske ressurser til å gjøre begge deler innenfor et enkelt tverrfaglig prosjekt. Og for det andre viser dette prosjektet at både faglige og samfunnsmessige nyttige resultat kan utvikles i et prosjekt som prioriterer spisset problemløsning over bred faglig kommunikasjon.

Problemet med å skrive på egne premisser er at forskningen aldri vil se dagens lys. Perspektiv og faguttrykk vil brukes på helt nye måter, og forskningsmiljøet rundt det tverrfaglige prosjektet vil stå uten mulighet til å forstå det faglige innholdet. Forskeren retter seg mot seg selv og en hypotetisk leser som deler hans forkunnskaper og interesse i å løse problemet. Forskeren gir blaffen i at det ikke finnes virkelige

lesere og unndrar prosjektet fra enhver kommunikasjon. En slik radikal orientering vekk fra forskerkolleger i den virkelige verden og mot en hypotetisk og i beste fall framtidsrettet lesergruppe er ikke akseptert i academia.

Ingen av disse valgene er særlig attraktive, og jeg mener derfor at dagens akademiske virkelighet med sitt ensidige fokus på publisering i disiplinorienterte tidsskrift, er en ikke-omgivelse for tverrfaglig forskning. Tematiseringen av tverrfaglige språkbruks- og kommunikasjonsproblemer er med på å synliggjøre dette underliggende problemet (Mathisen 1997, Evensen 2002a), men så lenge kravet om å dele kunnskap veier så tungt i forhold til å løse problemer, vil problemorientert, tverrfaglig og spesialisert forskning lukes ut av den akademiske hagen selv når de slår ut i blomst.

For meg er tverrfaglighet et helt virkelig problem. Som forsker og hovedfagsstudent samhandler jeg med forskere innenfor etablerte disipliner samtidig som jeg arbeider og samarbeider i et tverrfaglig prosjekt. Sett i forhold til hverandre er mine problemer knyttet til det tverrfaglige samarbeidet og syntese av fremmedfaglige perspektiv "bare barnemat" sammenlignet med oppgaven om å presentere dette tverrfaglige arbeidet ovenfor en disiplinorientert omgivelse. Fokuset på tverrfaglige problemer som indre samarbeidsproblemer som debatten ved NTNU tar utgangspunkt i (jf. Sørensen 1997, 2002), samstemmer derfor ikke med mine erfaringer fra prosjektet. Disse har vi perspektiv og verktøy for å løse. Jeg mener derfor debatten om tverrfaglighet i større grad bør fokusere på ytre evaluering av den enkelte forskeren og det enkelte tverrfaglige prosjekt (jf. Mathisen 1997). Dette mener jeg vi ikke har perspektiv og verktøy for å løse.

1.5 SjangDAI - et "tverr faglig" prosjekt

SjangDAI - En problemorientert, flerfaglig samarbeidsgruppe

Vår tverrfaglige prosjektgruppe, SjangDAI, ble dannet høsten 2000 av Arild Strømhylden og meg selv. Da prosjektet ble dannet, studerte Arild datateknikk og jeg anvendt språkvitenskap. Vi hadde derfor begge en fortid og framtid som spesialister på hvert vårt fagområde, og som to spesialister fra to forskjellige fagfelt, etablerte vi en flerfaglig ramme i prosjektet.

Innenfor denne flerfaglige rammen delte vi en felles nysgjerrighet for multiagentsystem og en felles *problem- og løsningsorientering*. Multiagentsystem, slik vi tilnærmet oss problemet, var et datasystem satt sammen av mange små programmer som gjennom kommunikasjon og samhandling løste komplekse problemer. Slik vi så det, falt dette problemet både utenfor og innenfor våre to faglige kompetanseområder. Datateknikk hadde kompetanse på utvikling av datatekniske systemløsninger generelt, men manglet kompetanse for de spesielle

kommunikasjonsproblemene som kjennetegnet multiagentsystem. Anvendt språkvitenskap hadde kompetanse på kommunikasjonsproblem, men manglet kompetanse om datatekniske system som var absolutt nødvendig for å løse et slikt problem. *Å løse kommunikasjonsproblem ved utvikling av et multiagentsystem* virket derfor som et ideelt, felles mål og felles problemstilling for vårt flerfaglige prosjekt.

Vårt mål var altså å samkjøre vår kommunikasjonsforståelse i en systemløsning, og vår språkvitenskapelige og datatekniske kompetanse skulle derfor veves tett sammen. Vi bestemte oss derfor for *ikke* å stykke opp ansvaret for sentrale deloppgaver knyttet til både problemene og løsningen, men å sammen dele dette ansvaret. Vi etablerte således et *felles engasjement* for å løse de forskjellige deloppgavene på best mulig måte, og prosjektet dannet dermed en tverrfaglig ramme både for gruppa som helhet og for den enkelte deltageren.

Det felles engasjementet for deloppgavene skapte en sterk lojalitet innad i gruppa. Men dette var ikke helt uproblematisk. Datatekniske, språkvitenskapelige og prosjektets behov gikk ofte på tvers av hverandre, og fordi vi var sammen om å løse de forskjellige deloppgavene, kunne vi ikke bare enes om å være uenige og arbeide hver for oss. Det felles engasjementet krevde derimot at tverrfaglige konflikter som dukket opp underveis, måtte gripes fatt i og løses. Gruppas faglige behov ble derfor tidlig prioritert høyere enn eventuelle motstridende språkvitenskapelige eller datatekniske behov. Vårt felles problem- og løsningsansvar var således en forenende drivkraft som etablerte et felles engasjement for deloppgavene.

Tverrfaglig utadvendthet og læring på tvers av datateknikk og språkvitenskap

Det felles engasjementet kombinert med erkjennelsen av det andre fagfeltets kompetanse gav prosjektet en spesiell ytre orientering. For det første vendte det felles engasjementet for deloppgavene oss innad mot samarbeidet og gruppeprosessene. Arild og jeg diskuterte faglige problemer oftere med hverandre enn med våre respektive fagmiljø, og vi fokuserte på at den andre i gruppa skulle forstå hva vi foretok oss. For det andre førte erkjennelsen av det andre fagfeltets kompetanse til at vi vendte oss utad mot hverandres fagfelt i vår personlige, faglige utvikling. Problemerkene som møtte meg, var f. eks. langt oftere datatekniske enn språkteoretiske, og jeg brukte derfor mye tid på å lære datateknikk. Disse innadvendte og utadvendte drivkreftene forsterket også hverandre, og samarbeidsprosessene som opprinnelig var rettet mot å løse felles problem, ble også ofte en arena for tverrfaglig læring.

Vår innadvendte samarbeidsform og utadvendte faglige orientering var problematisk. Jeg var ofte mer opptatt av å lære datatekniske problemer og forståelsesmåter enn språkvitenskapelige. Arild på sin side valgte ofte tekniske løsninger som var tilpasset

mitt kunnskapsnivå snarere enn forskningsfronten i sitt fagfelt. Vi sluttet dermed på mange måter å måle vår personlige kunnskapsutvikling som datatekniske eller språkvitenskapelige spesialister. Som en konsekvens av at vi valgte å åpne opp for tverrfaglig samarbeid, havnet derfor vår dialog og samarbeid med fagfeltene våre i andre rekke. Vår tverrfaglige utadvendthet og samarbeidsvilje fortonet seg raskt som en skjult arbeidsform og innadvendt faglig utvikling i forhold til anvendt språkvitenskap og datateknikk.

Sett utenfra kan vår tverrfaglige læring virke overflødig. Som gruppe disponerte vi den nødvendige kompetansen for å mestre de konkrete oppgavene. Jeg kunne f. eks. bare be Arild løse mine datatekniske problemer og forbli en rendyrket språkviter. Men så enkelt var det ikke. Dersom vi hadde ”abonnert” på slike faglige spesialtjenester hos hverandre, ville vi manglet både det nødvendige faglige blikket og fagspråket som vi trengte for å henholdsvis oppdage og kommunisere datatekniske og språkvitenskapelige problemer fra hver vår kant. Mangelen av felles fagspråk og fagkompetanse ville også hindret oss i å utveksle slike tjenester og forstå implikasjonene av de spesialfaglige tjenestene og løsningene vi ville utvekslet. Vi ville heller ikke kunnet vurdere om våre problembeskrivelser og løsninger lot seg kombinere. Konsekvensen dersom vi ikke hadde utviklet faglig innsikt på tvers av faggrensene, kunne i vårt tilfelle ha ført til at vår kommunikasjonsforståelse og våre datatekniske løsningsmodeller ikke hadde blitt samkjørt (jf. Java vs. Prolog s.33).

Vi trengte derfor å utvikle en gjensidig, tverrfaglig kompetanseplattform med et felles fagspråk for å mestre de forskjellige deloppgavene. Dette krevde at vi utviklet et felles blikk for problem og løsninger, og tverrfaglig læring var derfor en nødvendig del av begges individuelle arbeid. Denne nytten av tverrfaglig læring drev oss begge til å studere fag fra den andres fagretning. Disse studiene gav oss innsikt i hverandres fagspråk, faglige blikk og motivasjonen bak den andres faglige tilnærming. Den individuelle nytten av tverrfaglig læring var også en viktig motivasjon for samarbeidet. Når vi samarbeidet om problemløsingen, kunne vi bruke hverandre som faglige veiledere, og konkrete samarbeidsprosesser var derfor viktige arenaer for tverrfaglig læring.

En slik kombinert samarbeids- og læringsprosess var ”pair programming” (XP 2003). ”Pair programming” er en samarbeidsform der to personer sitter ved *samme* datamaskin og bytter på å skrive programkoden ved tastaturet og å kommentere det som skrives. Gjennom å samskrive programkode på denne måten kunne jeg både se Arild programmere, stille spørsmål og selv skrive kode med umiddelbar veiledning, råd og tilbakemelding fra en erfaren programmerer. Dette samarbeidet var en svært god arena for meg å lære datateknikk på, og jeg tenker med glede tilbake på denne perioden.

Etter hvert som mine ferdigheter i programmering utviklet seg, ble dette samarbeidet også et nyttig verktøy for å utveksle og utvikle språkvitenskapelig innsikt. Gjennom et felles programmeringsteknisk "språk" og metaforer kunne jeg synliggjøre språkvitenskapelige perspektiv og deres føringer i en felles virkelighet (jf. Evensen 2002a: 136). De indre sammenhengene i programmering fungerte som et medium for mine språkvitenskapelige syn på verden, og gjennom å velge ut, sette sammen og endre på programmeringsmetaforer, kunne vi sammen beskrive og modellere problem og løsninger på en måte vi begge forstod. Ved å programmere syntetiserte vi slik vår språkvitenskapelige og datatekniske fagspråk og kompetanse.

Både lærings- og problemløsningsprosessene gikk dermed i ett. Men dette betyr ikke at prosjektet først og fremst var en læringsprosess. Arbeidet med problemløsingen medførte riktignok læring, men læringsutbyttet var hele tiden sekundært i forhold til løsningen gjennom hele prosjektet. Problemet med multiagentsystem som har drevet prosjektet fram, er også et virkelig problem både faglig og for samfunnet. (Sam)arbeidsprosessene i dette prosjektet er derfor mer læringsbasert problemløsning enn problembasert læring.

Tilpasning av datatekniske og språkvitenskapelige perspektiv

Det felles engasjementet i konkrete deloppgaver krevde at vi ofte måtte bruke datateknisk og språkvitenskapelige perspektiv samtidig. Dette forutsatte at våre tverrfaglige perspektiv var samkjørte, og dersom de ikke var det, måtte vi tilpasse dem hverandre. Jeg vil nå gjennomgå to slike tverrfaglige tilpasningsprosesser i prosjektet vårt: en overordnet syntese mellom problem- og løsningsorientering og en syntese mellom en språkvitenskapelig teori og et datateknisk metodevalg knyttet til en spesiell deloppgave.

Syntese av språkvitenskapelige problem og datatekniske løsninger

Som nevnt hadde prosjektet en felles, overordnet problem- og løsningsorientering da det startet, men innholdet i dette fokuset var uklart. Målet om å utvikle et kjørbart datasystem var hentet fra datateknikk, og denne løsningsorienteringen la føringer for hvordan forskjellige problem skulle tilnærmes. I utviklingen av kjørbare datasystem blir ofte komplekse, tekniske problemer absolutte hinder som prosjektet er nødt til å løse. Tekniske kjerneproblemer kan derfor ikke utsettes til slutten av prosjektet fordi de da lett kan føre til at løsningen blir umulig å realisere og at prosjektet dermed ikke når sitt mål. Med et slikt datateknisk utgangspunkt, er det derfor formålstjenlig å gi tekniske problem en sentral plass også i prosjektets problemorientering.

Anvendt språkvitenskap har derimot andre erfaringer fra problem- og løsningsorientert arbeid med kommunikasjon. Disse erfaringene advarer mot å la

fokus på løsningen dominere arbeidet for tidlig. Et for sterk løsningsfokus vil kunne forenkle forståelsen av problemet slik at denne blir misvisende. Dette kan i sin tur føre til at forskere ender opp med å løse et annet problem enn det de egentlig var ute etter, og anvendt språkvitenskap mener derfor at et problemperspektiv bør legge føringer for løsningsorienteringen og ikke omvendt.

Innholdet i den datatekniske løsningsorienteringen kolliderte dermed med det språkvitenskapelige problemperspektivet, og det utviklet seg derfor raskt samkjøringsproblemer mellom disse to tilnærmingene i gruppa. Som språkviter var det min oppgave å forstå problemet først, og når jeg brukte mine spesialiserte fagbriller, anså jeg det derfor som konstruktivt å problematisere. Sett fra et datateknisk ståsted derimot virket denne spesialistrollen kverulerende og lite samarbeidsvillig. På sin side var dataspesialisten Arild primært opptatt av den kjørbare løsningen, og for ham var det derfor viktig å foreslå og snakke om forskjellige delløsninger. Fra mitt språkvitenskapelige ståsted virket dette derimot som om Arild til stadighet ”jumped the gun” og la villedende føringer for problemet.

Problemet med denne tverrfaglige konfliktene var at vi selv ikke var klar over dette da vi begynte samarbeidet. Vår forskjellige faglige orientering var en del av vår tause kunnen og kom først til syne i *måten* vi arbeidet og pratet fag på. Men etterhvert som samarbeidet utviklet seg, ble vi mer kjent med hverandres arbeidsform og mer bevisst våre egne faglige roller i samarbeidet. Denne økende bevisstheten gjorde det mulig for oss å diskutere forskjellene mellom våre forskjellige fagbriller og fagspråk. Vi kunne dermed si ifra når vi opplevde oss selv eller den andre som lite konstruktiv, og vi kunne gjennom slik metakommunikasjon utdype vår innsikt i både den andres og våre egne faglige perspektiv og tause kunnen om problemløsning. Denne metakommunikasjonen var avgjørende for både vårt tverrfaglige samarbeidsklima og vår personlige faglige utvikling.

Problem- og løsningsorienteringen vi endte opp med, ble derfor en hybrid som sprang ut fra både anvendt språkvitenskap og datateknikk. I det felles problem- og løsningsperspektivet valgte vi å beholde den positive tilnærmingen til komplekse kommunikasjonsproblem og den tekniske løsningens virkelighet. Fra den ene kanten ble jeg etter hvert overbevist om at datatekniske løsninger la føringer gjennom hele prosjektet og at det var nødvendig å *tenke med løsningene* hele veien. Fra den andre kanten ble Arild etter hvert overbevist om at kommunikasjonsproblem også la grunnlaget for løsningen og at det derfor var nødvendig å *tenke med problemene* hele veien. Men vi endret også disse perspektivene når vi tilpasset dem til hverandre. Fra min kant gikk vi bort ifra den humanistiske trangen til å grave i problemene. Jeg innså at jeg som humanist av og til hadde en tendens til å begynne en endeløs nøsting i problemene bak problemene bak problemene. Fra Arilds kant forlot vi den tekniske

tendensen til å velge ut problem ut fra løsningsalternativene. Arild innså at tekniske løsninger til tider kunne gi inntrykk av å basere seg på en bedre problemforståelse enn det var grunnlag for. Vi orienterte oss derfor både med og mot de datatekniske og språkvitenskapelige orienteringene samtidig.

Tilpasning av en datateknisk løsning: Java vs. Prolog

En annen tilpasning av tverrfaglige perspektiv skjedde i forbindelse med valget av en konkret datateknisk løsning for et bestemt språkvitenskapelig problem. Problemet som skulle løses, var hvordan agentene skulle tolke enkeltytringer (jf. s. 95ff), og arbeidet med å forstå dette problemet og velge en teknisk løsningsmodell var en viktig deloppgave i prosjektet.

Dette valget utkrystalliserte seg som et valg mellom Java og Prolog, og det var flere gode faglige argumenter for begge alternativene. Arild mente at Prolog var det beste alternativet. Innen datateknikk blir predikatlogikk som Prolog representerer, oppfattet som den beste løsningsmodellen til denne type problem, såkalt "best practice". Jeg derimot mente at vi burde bruke Java. Innen anvendt språkvitenskap assosieres predikatlogikk med en systemisk språkbruksforståelse, og dette regnes som "bad practice" for denne typen problem. Vi hadde begge sterke faglig interesser i dette spørsmålet, og ingen av de faglige argumentene var tunge nok til å overbevise den andre.

Valget om å bruke Java og ikke Prolog ble derfor først og fremst tatt av praktiske hensyn. Prosjektgruppa hadde allerede valgt å bruke Java som teknisk rammeverk for samhandlingsmiljøet til agentene, og jeg slet allerede med å lære meg Java og andre nødvendige datatekniske perspektiv i tilknytning til dette. Gruppa regnet det derfor som sannsynlig at jeg ikke ville greie å lære meg Prolog i tillegg, og i solidaritet med mine læringsutfordringer valgte Arild å sette sine datatekniske interesser til side. Dette var en sjenerøs handling og innebar at han måtte gi slipp på både egne læringsmuligheter tilknyttet Prolog og "best practice" i forhold til eget fagfelt og hovedfagsoppgave.

I etterkant av dette valget skjedde det en tverrfaglig syntetisering mellom den språkvitenskapelige forståelsen av agentenes ytringstolkning og de to tekniske rammeverkene. De to rammeverkene la forskjellige kommunikasjonsteoretiske forutsetninger til grunn for utformingen av tolkningskonteksten for ytringene i systemet. På den ene siden ville Prolog ha etablert et skille mellom run-time-miljøet for samhandlingen (Java) og run-time-miljøet for ytringstolkningen (Prolog). Et slikt skille ville gjort det lettere for oss å skille mellom problem knyttet til samhandling og problem knyttet til ytringstolkning. Men dette skillet bryter med den språkvitenskapelige tilnærmingen til samhandling og kommunikasjon vi valgte for

prosjektet. Denne forståelsen hadde vi på dette tidspunktet brukt i design av agentenes samhandlingsmiljø, og den forutsatte at tolkning av ytringer skulle basere seg på samhandlingskonteksten. Med utgangspunkt i kommunikasjonsproblemet var det derfor mer hensiktsmessig å forene kontekst for samhandling og ytringstolkning i ett run-time-miljø, snarere enn å skille samhandling og interaksjon fra hverandre i to run-time-miljø.

Skillet Prolog ville skapt mellom samhandlings- og tolkningskontekst er ikke teknisk umulig å gå utenom. Det er mulig å utvikle tekniske løsninger som ville gjort hele eller deler av samhandlingskonteksten tilgjengelig fra tolkningskonteksten og omvendt. Men et slikt skille mellom to forskjellige run-time kontekster ville gjort slike koblinger mye vanskeligere å etablere, og dersom disse koblingene var sentrale, slik vår språkvitenskapelige kommunikasjonsteori la opp til, kunne bruken av Prolog skapt problemer senere i prosjektet. Men på den annen side er det flere grunner til at Prolog regnes som "best practice" på dette problemområdet. Predikatlogikk gjør det lettere å konstruere tolkningsregler som kunne tilrettelagt for en historisk tolkningskontekst. Slike tolkningsregler vil være vanskeligere å etablere i Java. Begge løsningsalternativene ville dermed hatt flere positive og negative effekter på problemet, og det er derfor i etterkant vanskelig å entydig forsvare eller forkaste valget vi gjorde.

Etablering av en ny, felles, tverrfaglig og spesialisert plattform

Slik tverrfaglig syntetisering som er eksemplifisert over, hadde ikke bare konsekvenser for det interne faglige samarbeidet. Tilpasning av datatekniske og språkvitenskapelige perspektiv påvirket også forholdet til våre opprinnelige spesialfelt. Jeg vil her derfor presentere to eksempler på hvordan interne tilpassingsprosesser gjorde at prosjektet vårt mistet fotfestet i våre to opprinnelige spesialfelt.

Systemisk eller kommunikativt syn på programmering

I prosjektet vårt utviklet vi også et tverrfaglige perspektiv på programmering. Til å begynne med baserte vi oss på den rådende tilnærmingen til programmering innen datateknikk, her kalt systemutvikling. Systemutvikling tar utgangspunkt i de konstante aspektene ved teknologi og programmering, og fokuset er rettet mot hvordan menneskelige handlinger best mulig kan struktureres ut fra disse (Ørstavik 2001). Teorier for å forstå programmering og programmeringsmetoder er derfor ofte deklorative, og systemutvikling deler gjerne programmering inn i forskjellige faser og aktiviteter og lager regler for hvordan f. eks. tekniske, logiske og matematiske prinsipper enklest kan gjenskapes av systemutvikleren. Denne tilnærmingen er derfor først og fremst systemisk (jf. s. 46): teknologien bruker systemutvikleren som et

medium å realisere seg gjennom, snarere enn at programmereren og mennesket bruker teknologien som et medium å utfolde seg i.

Fra språkvitenskapelig side, hadde vi svært få direkte føringer for programmering. Teknisk tunge aktiviteter anses av humanister gjerne som teknologens ansvarsområde, og anvendt språkvitenskap utfordret derfor ikke den teknologiske, systemiske forståelsen av programmering som en ikke-menneskelig aktivitet. I de to spesialfeltene rådet derfor en systemisk oppfatning om at teknologien, og ikke utvikleren, legger premissene for programmering.

Underveis i prosjektet kom vi i kontakt med flere grasrotgenererte datatekniske metoder som ”extreme programming” (XP 2003) og ”design patterns” (Gamma 1994). Vi oppdaget at disse metodene hadde mye til felles med moderne språkvitenskapelige kommunikasjons- og språkbruksteori (Ørstavik 2001). Da vi i den tverrfaglige gruppa etter hvert lærte oss både programmering og språkvitenskap, oppdaget vi at programmering kunne forstås som språkbruk på lik linje med andre former for kommunikasjon. Denne tverrfaglige kombinasjonen av praktisk programmering og språkvitenskapelig forståelse av språkbruk viste seg å gjøre våre egne programmeringsaktiviteter mer tilgjengelig både praktisk og teoretisk. Dette skulle vise seg å bli sentralt i mitt delprosjekt.

Gradvis forstod vi dermed programmering først og fremst som sosial, dynamisk og kreativ språkbruk. Dette perspektivet har spesialiserte røtter fra både datateknikk og anvendt språkvitenskap, men det bryter også tvert med den etablerte forståelsen av programmering innen systemutvikling. Som helhet er dette sentrale perspektivet historieløst og fremmed for både språkvitere og datateknikere, og det har derfor bare delvis forankring i sine opphavelige spesialfelt. Dette perspektivet er beskrevet i neste kapittel.

Forskjellige problemstillinger tilknyttet multiagentsystem

Et annet eksempel på et tverrfaglig spesialisert perspektiv er vårt problem- og løsningsperspektiv på multiagentsystemer. Multiagentsystemer er et etablert problem innenfor datateknikk (Jennings 2000), og dette problemet er etablert i en større pragmatisk og strukturalistisk tradisjon. Spørsmålene forskerne arbeider med omkring dette problemet er derfor først og fremst knyttet til interaksjon, språk og handling som struktur: ontologi, semantikk, pragmatikk, generativ logikk, strategi hierarki, intensjonsnettverk m. fl. Tradisjonen forutsetter dermed at en systemisk kommunikasjonsforståelse (jf. s. 46) er best egnet i multiagentsystem, og agentenes individuelle egenskaper er ment å generere handlinger som deretter skaper samhandling.

I vårt prosjekt har vi basert oss på en helt annen kommunikasjonsteoretisk tradisjon (jf. s. 54). Vi har bakgrunn fra en interaksjonistisk og konstruksjonistisk tradisjon, og vi ser derfor andre problem og stiller andre spørsmål i forhold til multiagentsystem. Bl. a. oppfatter vårt sosio-interaksjonistiske perspektiv individuelle tanker og handlinger mer som samhandlingsgenerert enn omvendt, og individuell, intern tolkning mer som ytre, kontekstuell drevet enn omvendt. Agentene generer dermed ikke så mye handlinger og kommunikasjon som samhandling, kommunikasjon og agentene generer hverandre gjensidig. Sammenlignet med den systemiske tilnærmingen til multiagentsystem er vi derfor mer opptatt av hvordan agentene etablerer og deretter gjør bruk av en dynamisk, spesiell kontekst, snarere enn å forstå hvordan agentene bruker en statisk, generell kontekst for deretter å generere tekst.

Vårt overordnede mål om utvikling av multiagentsystem overlapper med det overordnede målet i det etablerte fagfeltet rundt multiagentsystem, men de konkrete problemstillingene vi har valgt å ta tak i faller ofte på siden av fagfeltets problem- og løsningsfokus. Våre problemstillinger knyttet til kontekst, kontekstbasert tolkning, samhandling, språkbruk m.m. virker løsrevet fra fagfeltet fordi de baserer seg på kommunikasjonsteoretiske forutsetninger som er grunnleggende fremmede for en systemisk tradisjon. For mer om forskjellen mellom vår tilnærming og tradisjonelle tilnærminger til multiagentsystemer se Strømhylden (2002: 18-26, 48-68).

SjangDAIs tverrfaglige kommunikasjon

Eksemplene over viser hvordan våre problem og vårt samarbeid drev oss til å utvikle en egen, spesialisert tverrfaglig plattform. Denne plattformen er fremmed sett fra våre opprinnelige fagfelt, og disse fagfeltene kan derfor lett oppfatte forskningen vår som historieløs i henhold til deres referanserammer. Men vi har heller ikke funnet noen andre fagfelt eller kommunikasjonsarenaer som har en dypere forståelse av prosjektets bakgrunn eller mer sammenfallende kunnskapsinteresser. Til tross for mangelen av felles historiske referanserammer er derfor våre opprinnelige fagfelt fremdeles de best egnede målgruppene, og når vi skal dele vår tverrfaglige kunnskap med disse målgruppene, trenger vi derfor å forstå slike tverrfaglige kommunikasjonsproblem.

Forskningen i vårt prosjekt er utført på en tverrfaglig plattform, og jeg trenger derfor å basere meg på dennes forutsetninger når jeg skal diskutere problem og resultat som vi har utarbeidet på denne plattformen. Samarbeidet, den tverrfaglige læringen og bruken av taus og uttalt datateknisk og språkvitenskapelig kunnen var et nødvendig fundament for arbeidet med å forstå problemene og utvikle løsningen, og det er derfor nødvendig at andre som ønsker å forstå dette arbeidet også trenger å basere seg på dette fundamentet. Når jeg som skriver skal presentere, beskrive og forklare vår forskning, trenger jeg derfor også å basere meg på prosjektets tverrfaglige

forkunnskap og kunnskapsinteresser. Disse forutsetningene er bakgrunnen og premissene for mine faglige poeng.

Våre målgrupper innenfor anvendt språkvitenskap og datateknikk deler derimot ikke disse forkunnskapene og kunnskapsinteressene. Faglig kommunikasjon innenfor disse disiplinene er vant til å ta utgangspunkt i andre forutsetninger. Lesere fra disse disiplinene leser basert på forkunnskaper og kunnskapsinteresser etablert i disiplinen. ”Å lese på skriverens premisser” blir tomme ord dersom leseren ikke har kjennskap til disse premissene. Dette gjelder i kommunikasjon mellom vårt tverrfaglige prosjekt og våre disiplinorienterte omgivelser. Som skriver av mitt tverrfaglige prosjekt kan jeg derfor ikke uproblematisk forutsette at mine premisser er kjent og bruke felles historiske referanserammer for å synliggjøre denne for leseren; som tverrfaglig skriver er jeg nødt til å etablere denne plattformen av forkunnskaper og kunnskapsinteresser dersom jeg ønsker å diskutere prosjektets forskning utført på denne plattformen. Denne etableringen av felles forkunnskaper og kunnskapsinteresser basert på få felles historiske referanserammer er svært problematisk, og jeg vil her beskrive hvordan denne problematikken har påvirket tidligere kommunikasjon mellom vårt prosjekt og fagfeltet anvendt språkvitenskap.

Erfaringer fra tidligere kommunikasjon med anvendt språkvitenskap

Som språkviter har jeg allerede levert tre tekster om prosjektet til anvendt språkvitenskap. Tilbakemeldingen på disse tekstene har variert endel, og disse erfaringene vil jeg her bruke som utgangspunkt for å diskutere de tverrfaglige kommunikasjonsproblemene for vårt prosjekt.

Erfaringene fra tilbakemeldingene er at når jeg som skriver tar utgangspunkt i fagfeltets forkunnskaper og kunnskapsinteresser, har lesere fra anvendt språkvitenskap framhevet prosjektet som interessant. Beskrivelser av datatekniske perspektiv som baserer seg på fagfeltets forkunnskaper, har derfor ofte fått positiv tilbakemelding. Leserne mine har også ofte oppfordret meg til å utdype problemstillinger og spørsmål som er relevante for fagfeltets kunnskapsinteresser.

Problemet med denne tilbakemeldingen er at den ofte har kommet uavhengig av om disse problemstillingene også er relevante i forhold til prosjektets målsetninger. Kunnskapsinteressene til fagfeltet har ofte ligget på siden av prosjektets kunnskapsinteresser, og diskusjoner relevant for fagfeltet har ofte vært mindre relevante for prosjektet. Positiv tilbakemelding som tar utgangspunkt i fagfeltets virkelighet, har derfor en tendens til å virke litt lettkjøpt når vi møter en helt annen virkelighet i prosjektet.

Når vi har diskutert tema med utgangspunkt i prosjektets kunnskapsinteresser og forkunnskaper, har tilbakemeldingene derimot vært mindre positive. Problemstillinger og spørsmål som er direkte relevante for prosjektets problemer og løsninger, har ofte blitt avfeid som uforståelige eller lite faglige. På meg har det virket som om datatekniske innslag kjedet leseren og bare tolerert når de enten ble begrenset, underordnet språkvitenskapelige interesser eller skilt ut som sidesprang. Prosjektets tverrfaglige interesser har slik blitt veid og ofte blitt funnet for lett sammenlignet med fagfeltets interesser. Dermed har mye av samarbeidet, den tverrfaglig læringen og bruk av datateknisk kompetanse blitt oppfattet som irrelevant og ikke verdsatt.

Når jeg som skriver ønsker å presentere og dele forskningen vår med mitt språkvitenskapelige fagmiljø, kommer jeg ikke utenom den datatekniske og tverrfaglige bakgrunnen i prosjektet. Den tverrfaglige bakgrunnen er mine premisser som skriver. Men som tidligere tilbakemelding viser, etablerer fagfeltets andre forkunnskaper og kunnskapsinteresser som premisser for sine lesere. Resultatet av dette er at de tverrfaglige tekstene jeg har skrevet har blitt lest som disiplinorienterte. Dette kan virke uunngåelig, og leseren kan ikke bruke og basere seg på forkunnskaper, kunnskapsinteresser og andre premisser han ikke har. Men den tverrfaglige skriveren kan heller ikke løsrive seg fra de tverrfaglige forutsetningene og premissene i prosjektet. Dermed stopper kommunikasjonen opp.

Umiddelbart kan det virke som den eneste løsningen på dette problemet, er at den tverrfaglige skriveren kommer den disiplinorienterte leseren i møte. Som denne hovedfagsoppgaven forhåpentligvis vil vise, er det mulig for et tverrfaglig prosjekt å gå et disiplinorientert fagfelt i møte og greie å synliggjøre en tverrfaglig bakgrunn og forskning også på tvers av forskjellige faglige kunnskapsinteresser og forkunnskaper. Men det er også grenser for hvor mange forkunnskaper, kunnskapsinteresser og forklaringer som kan utdypes uten at prosjektet mister kontakten med sine opprinnelige forutsetninger og interesser. Derfor kan ikke disiplinorienterte lesere forvente at tverrfaglige skrivere skal forlate sin egen, spesialplattform og samtidig greie å dele sine resultater. Mottakere av tverrfaglig utviklet kunnskap må akseptere at tverrfaglig forskning svært ofte *må* beskrives mot en tverrfaglig bakgrunn. Konklusjonen blir derfor at akademia ikke bare kan forlange av den tverrfaglige skriveren å tilpasse seg den disiplinorienterte leserens premisser: den disiplinorienterte leseren må også tilpasse seg den tverrfaglige skriverens premisser.

Den tverrfaglige skriverens premisser

Hva innebærer så det å tillegge den tverrfaglige skriverens premisser større vekt i kommunikasjonen? For det første trenger den tverrfaglige skriveren å bli gitt friere rammer til å etablere kunnskapsinteresser som også leseren er villig til å dele. Vårt tverrfaglige prosjekt har f. eks. en problem- og løsningsorientering som ligger på

siden av fagfeltets. Vår kunnskapsinteresse er derfor ganske fremmed for anvendt språkvitenskap, men for å kunne dele og kommunisere kunnskap produsert basert på denne motivasjonen, trenger jeg til en stor grad å forutsette denne interessen. Skal lesere fra anvendt språkvitenskap få del i våre tverrfaglige erfaringer, må de også være villige til å dele den motivasjonen som ligger bak handlingene som skapte disse erfaringene.

Sammenlignet med mer disiplinorienterte prosjekt som ofte baserer seg på kunnskapsinteresser allerede etablerte i fagfeltet (ytre validitet allerede etablert i fagfeltet), må derfor tverrfaglige prosjekt i langt større grad begrunne og rettferdiggjøre en ukjent kunnskapsinteresse. I vårt prosjekt kan jeg derfor ikke forutsette at leseren oppfatter vårt prosjekt som relevant og nyttig, men jeg trenger å etablere denne felles interessen for problem- og løsningsorienteringen i kommunikasjonen. Denne oppgaven kan være svært omfattende, og som vårt prosjekt viser, kan denne kunnskapsinteressen også basere seg på historiske referanserammer som går på kryss og tvers av faggrensene. Motivasjonen for utvikling av multiagentsystem er f. eks. etablert innen datateknikk, mens validiteten av valget av en sosio-interaksjonistisk tilnærming til kommunikasjon er etablert i anvendt språkvitenskap.

Som beskrevet i eksemplene om forskjeller mellom våre perspektiv og deres opphav fra disiplinene, kan det være helt nødvendig å basere seg på et sammensurium av datateknisk og språkvitenskapelig kompetanse når et slikt tverrfaglig fokus etableres. Dersom alle tverrfaglige referanser skal forklares og utdypes for den disiplinorienterte leseren, vil oppgaven med å etablere en felles kunnskapsinteresse i praksis bli skriverens. Lesere fra disiplinene må derfor forvente at den tverrfaglige skriveren etablerer en ukjent kunnskapsinteresse, men leseren må også tolerere at den tverrfaglige skriveren baserer seg på tverrfaglige, historiske referanserammer som går utenfor leserens faglige forkunnskaper. Referansene kan derfor ikke bare forstås som gjenkjennelsesmarkører, men leseren må forholde seg til dem som aktive underbygninger i teksten.

Tverrfaglige skrivere trenger derfor i langt større grad enn disiplinorienterte skrivere å basere seg på utadvendt og lærende faglig lesing. Leserens blir i langt større grad nødt til å enten stole på skriveren når det gjelder faglige vurderinger tatt på bakgrunn av en faglig spesialisering en selv ikke besitter, eller selv aktivt å gå skriveren i sømmene. Denne hovedoppgaven synliggjør nettopp dette. Som skriver kan jeg ikke redegjøre for f. eks. hele den datatekniske eller språkvitenskapelige spesialiseringen for lesere som ikke har kjennskap til disse spesialistområdene. Må jeg som skriver gjøre det, vil jeg bli presset til å skrive en lærebok om begge fagfeltene som forord til denne teksten. Jeg kan heller ikke unnlate å trekke argumentasjonskjedene mine inn i disse

spesialiserte landskapene, da svært mye av forskningen vår er utført nettopp innenfor disse virkelighetene. Jeg er derfor nødt å referere også til spesialiserte rammer, selv om disse er fremmede for disiplinorienterte lesere.

1.6 Konklusjon: løsning på tverrfaglige kommunikasjonsproblem

Tverrfaglig lesing

Ønsker vi å forstå og dele tverrfaglig forskning, mener jeg vi er nødt til å akseptere den som tverrfaglig. Tverrfaglig kommunikasjonsproblem som skissert over, må derfor løses med et kompromiss mellom skriver og leser. Riktignok må det skrives på leserens premisser, men det må også leses på skriverens premisser (Rommetveit 1974 sitert etter Nystrand 1986). Dette medfører at vi i konkrete tverrfaglige skrive- og lesesituasjoner må forvente et betydelig arbeid ikke bare av den tverrfaglige skriveren, men også av eventuelle disiplinorienterte lesere. Med utgangspunkt i etablerte forventninger i dagens akademiske virkelighet (jf. Mathisen 1997), betyr dette at kommunikasjonen i større grad er nødt til å skje også på den tverrfaglige skriverens premisser. Hva innebærer det så å lese på en tverrfaglig skrivers premisser? Leseren må forvente at den faglige bakgrunnen som skriveren arbeider på for utenforstående kan virke både spesialisert og historieløs. Dette setter krav til leserens evne til å omstille seg, kanskje særlig når det gjelder de faglige interessene i kommunikasjonen. Leseren må også være klar over at skriveren må basere seg på historiske referanserammer som kan være fremmede, og som kan gjøre den faglige argumentasjonen vanskelig å følge. Spesialiserte tverrfaglige perspektiv lar seg ofte ikke forklare mot en enkelt disiplinorientert bakgrunn, og leseren må derfor sette seg inn i og lære disse faglige premissene. Vi trenger derfor ikke bare ”polyvalente” tverrfaglige forskere (Sørensen 2002), vi trenger også ”polyvalente” lesere.

Lesepraktiser der leseren i større grad forventes å lære og utvikle seg sammen med skriveren betrakter jeg som framtidrettede. Slik praksiser vil gjøre det mulig for tverrfaglige forskere å bruke gamle og velkjente faglige redskaper til stadig å pløye nytt og fremmed faglig terreng. Slik praksiser vil dermed slippe den tverrfaglige forskeren løs i framtida og frigjøre dem til å samarbeide og arbeide problemdrevet og læringskrevende. Tverrfaglig lesing må speile tverrfaglig skriving, og begge praksisene trenger hverandre. Det stilles større krav til at en leser av en tverrfaglig, historieløs, spesialisert fagtekst viljer og evner å innta nye standpunkt, etablere nye interesser og lære.

Konsekvenser for resten av denne oppgaveteksten

I denne oppgaven vil jeg derfor henvende meg til en tverrfaglig leser fra anvendt språkvitenskap. Jeg vil ikke forvente at denne leseren vil dele min tverrfaglige og datatekniske bakgrunn, men jeg vil likevel benytte meg av historiske referanser til

denne bakgrunnen. Mine resultat baserer seg på både på flere års spesialistutdanning innen datateknikk og to års tilpasninger av både datatekniske og språkvitenskapelige perspektiv. Dersom jeg skulle basert meg utelukkende på språkvitenskapelige referanserammer, ville jeg ha måttet skrevet våre forskningsresultater innenfor rammene av et svært undervisningsverk og gjort det til min oppgave å først utdanne leseren i datateknikk og deretter tverrfaglige tilpasninger av anvendt språkvitenskap og datateknikk. Dette ville være en urealistisk oppgave å gjennomføre innenfor rammene av denne ene teksten.

Jeg har derfor valgt å dele denne oppgaven inn i tre kapitler. Dette første kapitlet om tverrfaglighet er i dialog med ett fagfelt, nærmere bestemt diskusjonen om tverrfaglighet innenfor NTNU. Jeg regner derfor dette kapitlet som disiplinorientert, og ikke tverrfaglig. I andre kapittel vil jeg derimot gå inn i et sentralt perspektiv i den konkrete tverrfaglige bakgrunnen i mitt hovedfagsprosjekt. Her vil jeg presentere et nytt tverrfaglig metodisk perspektiv som vi har utviklet og brukt i dette prosjektet. Dette perspektivet gir en relativt god forståelse av vår tverrfaglige tilnærming. Denne teksten vil henvende seg til anvendt språkvitenskap, men forutsette en viss innsikt i programmering og IKT. Det tredje og siste kapitlet i denne oppgaven henvender seg til en tverrfaglig leser. I denne teksten vil jeg forlate svært mange faglige språkvitenskapelige forutsetninger og basere seg meg i utstrakt grad på både datatekniske og tverrfaglige forutsetninger. Dette kapitlet krever at leseren i stor grad er villig til og evner å leve seg inn i våre tverrfaglige forkunnskaper og kunnskapsinteresser.

”The moral of this book is: Check your premises” (Greenberg 1972).

Programmering som språkbruk Kapittel 2

2.1 Samspillet mellom programmering og programmeringsspråk

Programmering og programmeringsspråk

Programmering og programmeringsspråk⁵ bruker og former hverandre gjensidig. På den ene siden brukes programmeringsspråk i programmering, og språkets muligheter og begrensninger styrer programmereres valg. På den andre siden realiserer programmeringsspråk seg i programmeringsprosessene, og utviklingen av programmeringsverktøy og programmeringsspråk styres og utføres i programmering. Programmering og programmeringsspråk blir dermed sentrale premissleverandører for hverandre, og disse fenomenene bør derfor forstås i sammenheng.

Programmering og programmeringsspråk eksisterer videre på et større premissgrunnlag. Konkret bruk av programmeringsspråk utføres av personer som f. eks. ønsker å løse et praktisk problem eller lage et nyttig program. Programmering er dermed motivert og drevet i konkrete, levde situasjoner. Programmererne og deres situasjonelle betingelser er videre del av en større sosial omgivelse. Praktiske problem og gode løsninger er og blir delt med andre, og til og med selve bruken av programmeringsspråk skjer i spill med andre utviklere, enten direkte gjennom samarbeid om et konkret program eller indirekte gjennom bruk av andres program og system som forutsetninger for programmet. Programmering kan derfor forstås som situasjonelt og sosialt betinget bruk av programmeringsspråk (Ørstavik 2001).

De situasjonelle betingelsene påvirker også programmeringsspråket. Utvikling av gjenbrukbar kode er en sentral side ved bruk av programmeringsspråk: ”When [designers] find a good solution, they use it again and again” (Gamma 1994: 1). For å forenkle denne formen for gjenbruk av løsninger, bygges gode løsninger ofte inn i programmeringsspråket som leksikalske eller konvensjonelle ressurser (jf. biblioteket i Java, og ”design patterns” (ibid.)). Dermed er *både* bruk *og* (videre)utvikling av programmeringsspråk situasjonelt drevet.⁶

⁵ For en rask introduksjon til programmeringsspråk se Encyclopædia Britannica Online (2003).

⁶ I vårt prosjektet var det nettopp gjentatte og forventede behov i enkeltsituasjoner som drev fram utviklingen av både leksikalske, syntaktiske og konvensjonelle gjenbrukbare ressurser i vårt programmeringsspråk.

De situasjonelle kreftene som driver fram utvikling av programmeringsspråk, har også en sosial side. Ettersom mange programmerere møter lignende problem og behov, gjenbrukes de derfor også andres gode løsninger. Gode, gjenbrukbare løsninger distribueres og brukes derigjennom av mange forskjellige programmerere.

Oppbygningen av programmeringsspråk tilrettelegger også forskjellig for slikt samarbeid, og programmeringsspråks popularitet er derfor tett knyttet til hvordan det tilrettelegger for både at utviklere kan tilføre språket egne, gjenbrukbare løsninger (situasjonelt betinget tilpasning) og at de kan dele disse gjennom felles språklige ressurser eller verktøy (sosialt betinget utvikling).

Programmering som språkbruk

Programmering og programmeringsspråk former og formes av både hverandre og sine sosiale og situasjonelle omgivelser. I tillegg til dette er programmering også et symbolsk fenomen, og dette kan bl. a. spores i programmeringsspråkenes syntaktiske, leksikalske og konvensjonelle karakter. Bruk av programmeringsspråk er bruk av symbolske ressurser, og ettersom programmeringsspråk og programmering er situasjonelt og sosialt betinget, kan programmering forstås som situasjonelt og sosialt betinget bruk av symbolske ressurser.

Programmering som situasjonelt betinget bruk av symbolske ressurser blir synlig når programmererens situasjonelle motiv og behov utkrystalliseres i en konkret situasjon.

”Jeg bruker gjerne programmeringsspråket eller programmodeller for å utforme eller skape tanker eller ideer. [...] Programmeringsspråket er et medium eller et tankeverktøy (Hoel 2000) jeg bruker for å skape og forme tanker om og representasjoner av fenomener i verden” (Ørstavik 2001: 10).

Behov og motiv fra en konkret situasjon og symbolske ressurser fra programmeringsspråket utgjør dermed begge en ramme i programmererens perspektiv. Gjennom sin rolle som kokonstituent av programmererens perspektiv, muliggjør og begrenser slike språk programmererens valg og handlinger, og dermed er programmeringsspråket med på å bestemme hvilke fokus, behov og motiv programmererens perspektiv kan romme i de enkelte situasjonene. Programmering blir dermed en syntese av den konkrete situasjonen og det abstrakte, symbolske språket i programmererens perspektiv og handlinger (jf. Vygotsky 1978: 25f, ”unity of perception, speech, and action”), og bruk av symbolske, programmeringsspråklige ressurser blir dermed forankret i og betinget konkrete programmeringssituasjoner (Ørstavik 2001: 10f).

Den sosiale bruken av symbolske programmeringsspråk er mer iøyenfallende enn den individuelle bruken beskrevet over. Den sosiale siden ved programmeringsspråk er derfor også tydeligere tematisert i forskning om tema:

”The process provides interaction between users and designers, between users and evolving tools, and between designers and evolving tools. It is an iterative process in which the evolving tool itself serves as the medium for communication” (Baetjer 1998, sitert etter Pressman 2000: 17).

Programmerere kommuniserer med hverandre og brukere av ferdige program ved å bruke symbolske løsninger bygd inn i programmeringsspråket og programmet. Programmerere utveksler dermed tanker og løsninger konstruert i programmeringsspråket som gir mening for dem og brukere i forhold til deres situasjonelt betingede behov, motiv og perspektiv (Ørstavik 2001: 11ff).

Språklige forståelsesmodeller overført til programmering

Som vist over, kan forståelsesmodeller for både språk og språkbruk overføres til både programmeringsspråk og programmering. Den ene koblingen mellom språk og programmeringsspråk er allerede utbredt i både academia og samfunnet generelt, og begrep som ”språk”, ”syntaks”, ”bibliotek” o.l. brukes ofte som metaforiske bilder for å beskrive noen tekniske egenskaper ved programmering (jf. Lakoff 1980). Denne metaforiske bruken av ”språk” som forståelsesmodell for programmering kan virke allestedsnærværende og det er derfor ofte vanskelig å skille metaforen fra det opprinnelige fenomenet, også i forskningsøyemed.

Når det er snakk om programmeringsprosesser, aktiviteter og praksiser, forsvinner derimot de fleste metaforiske referanser til språk og språkbruk. Det er derfor ikke vanskelig å skille mellom metafor og fenomen når en språklig forståelsesmodell brukes om ikke-tekniske programmeringsfenomen. Programmering beskrives derimot med bl. a. naturvitenskapelige metaforer, og f. eks. brukes naturvitenskapelige begrep som ”evolusjon”, ”livssyklus” og ”debugging” om programmering. Disse metaforene gir inntrykk av at programmet og programmeringsspråket lever sine egne liv løsrevet fra programmereren. Andre tekniske begrep som ”kravspesifikasjon” og ”software” gir inntrykk av at programmering er tekniske eller økonomiske fenomen. Disse metaforene står på mange måter i motfase til en forståelse av programmering som språklige fenomen og kan innimellom til og med presentere menneskelige sider av programmering som et sidespor eller en feilkilde (jf. Pressman 2000, Jacobson 1998).

Mangelen av metaforiske koblinger mellom språkbruk og ikke-tekniske programmeringsfenomen, følger et etablert faglig mønster. Programmering oppfattes som tekniske fenomen, og kategoriseres derfor som et domene for ingeniør- og naturvitenskapene (jf. Schwebs 1994: 1f). Språkbruk derimot oppfattes som menneskelige fenomen og behandles som et humanistisk forskningsobjekt. Bruk av en språkbruksmetafor om programmering går dermed på tvers av anerkjente faggrenser og ikke minst faglig forståelse innen både språkvitenskap og datateknikk og bør behandles som tverrfaglig.

Til tross for manglende faglige bakgrunn og kjensgjerningen at språkbruk og programmering er to forskjellige fenomen med hver sin historie, kan det likevel være interessant å bruke en språkbruksmetafor om programmeringsprosesser, aktiviteter og praksiser. Som denne korte innledningen viser, er det likhetstrekk mellom programmering og språkbruk. For det første gjør både programmering og språkbruk krav på å være situasjonelt og sosialt betinget bruk av symbolske ressurser. For det andre er både språkbruk og programmering meningsfulle aktiviteter for deltakerne i forhold til deres sosialt situerte handlinger. Og for det tredje er både språk og språkbruk, programmeringsspråk og programmering avhengige av hverandre. Videre er en slik metafor ikke interessant i forhold til om den er riktig eller gal, men i forhold til om den gir relevant innsikt i fenomenet programmering. Dersom en språkbruksmetafor kan gi en nyttig metodisk innfallsvinkel til programmering, vil denne tverrfaglige koblingen ha faglig verdi. En slik metaforisk tilnærming kan også ha en speilende virkning tilbake på språkbruk, og den metaforiske koblingen vil kunne fungere som bro for å overføre innsikt fra språkbruk til programmering og tilbake igjen. Dersom denne metaforiske koblingen i tillegg tilrettelegger for bedre programmeringsprosesser, aktiviteter og praksiser, kan denne innfallsvinkelen også få anvendt verdi.

Plan for dette kapitlet

En forståelse av programmering og programmeringsspråk har vært en nødvendig forutsetning for vårt prosjektet. Målet for prosjektet var å løse praktiske problem ved utvikling av multiagentsystem, og som vi vil vise her, ligger dette problemet i skjæringslinja mellom programmering og programmeringsspråk. I denne skjæringslinja møtte vi noen store utfordringer, og før jeg kan ta for meg vår tilnærming til prosjektets hovedproblem, nemlig *utvikling av multiagentsystem*, trenger jeg først å gripe fatt i disse utfordringene tilknyttet *samspillet mellom programmering og programmeringsspråk*. I denne drøftingen vil jeg presentere en hypotese som utkrystaliserte seg underveis i prosjektet: teoretisk innsikt i språklig kommunikasjon har praktisk anvendelse både i programmering generelt og multiagentsystem spesielt.

Samspillet mellom programmering og programmeringsspråk krever dypere innsikt i disse fenomenenes relasjon til sine situasjonelle og sosiale betingelser. Jeg mener derfor at en språkbruksmetafor vil være nyttig som metodisk innfallsvinkel for å belyse programmering og programmeringsspråk. Innsikt i programmering og multiagentsystem er derfor motivasjonen bak arbeidet vårt, og språkbruk er hovedfokuset i denne teoretiske og metodiske diskusjonen om programmering.

For å belyse denne innfallsvinkelen vil jeg ta utgangspunkt i to forståelsesmodeller for språk og språkbruk: en systemisk og en dialogisk modell. Den systemiske modellen er som nevnt, allerede utbredt både som metafor for programmeringsspråk og danner både direkte og indirekte et grunnlag for å forstå programmering som et grunnleggende systemisk fenomen. Den dialogiske modellen er derimot fremmed for dette problemområdet. Jeg vil derfor gå grundigere til verks for å vise hvordan denne kan brukes som innfallsvinkel til studier av programmering og programmeringsspråk.

Formålet med beskrivelsen av den systemiske forståelsesmodellen for språk og dens metaforiske overføring til programmering, er ikke å gi et representativt bilde av moderne studier av språk eller en dekkende oversikt over etablerte tilnærminger til programmering. Denne beskrivelsen er først og fremst ment å hente fram noen mye omtalte programmeringsfenomen og eksemplifisere hvordan en språklig forståelse kan kobles til programmering.

2.2 En systemisk tilnærming til språk og programmering

En systemisk forståelse av språk

Språk er et svært omfattende fenomen som er av helt avgjørende betydning for alle menneskers liv og leven, og det eksisterer derfor svært mange forskjellige oppfatninger om språk. Før jeg kan gå inn og diskutere hvordan en systemisk forståelse av språk brukes metaforisk om programmering, trenger jeg derfor å redegjøre for hva jeg mener med en systemisk forståelse av språk.

En systemisk tilnærming ser språk som et system som eksisterer forut for språkbruk. ”Systemet” er her den bærende metaforen og forståelsesmodell, og systemet defineres først og fremst gjennom systemdelenes relasjon til hverandre *innen* systemet. Denne tilnærmingen ble først etablert av Saussure (1916) som beskriver språket som en samling sosialt skapte naturlover og ”a thing that can be found in the world of other real things” (Crowley 1990: 30). Senere forskere har bygget videre på denne tilnærmingen, men i stedet for å forstå språket som et system med opphav utenfor mennesket, forstås språkssystemet nå mer som et menneskelig og nevrologisk betinget fenomen: “Underlying principles of generative grammar [...] must be part of the intellectual organization which is a prerequisite for language acquisition [and use]” (Chomsky 1966: 42). Til tross for endringene i synet på språkets opphav, forstås språket fremdeles som et system forut for og uavhengig av språkbruk.

Ettersom språkssystemet eksisterer forut for språkbruk, utelukkes påvirkningen fra selve språkbruken på språkssystemet. Språkssystemet er alltid noe som er *gitt til* og ikke *skapt av* språkbrukerne, og språkssystemet blir derfor også tillagt en universell egenskap i forhold til språkbrukerne (Bakhtin 1981: 270, Crowley 1990: 43f): ”A

language is something which is in each individual, but is none the less common to all” (Saussure 1916: 19). Denne universelle egenskapen omfatter både individuelle personer og konkrete språkbrukssituasjoner. Språkssystemet fungerer dermed som grunnleggende premisser for språkbruk, en slags kommunikasjonens tyngdekraft, og påvirkningen mellom de to fenomenene skjer systematisk og ensidig *fra* språket *til* språkbruk.

En systemisk forståelse av språk kan også spores utenfor akademien. Reddy (her sitert etter Lakoff 1980: 10-13) beskriver bl. a. hvordan det engelske språket er fullt av metaforiske uttrykk som skildrer språk som system:

*”Our language about language is structured roughly by the following metaphor:
Ideas (or meanings) are objects.
Linguistic expressions are containers.
Communication is sending.”*

Denne ”conduit metaphor” (ibid.) beskriver både språk og mening som strukturer som kan knyttes til hverandre, og språkbruk er å innkode og avkode meningsobjekter fra de språklige strukturene samt å utveksle språklige strukturer som inneholder meningsobjekter mennesker imellom (jf. Saussure 1916: 11ff). Denne alminnelige, systemiske forståelsen av språk forutsetter dermed: 1) at språket er strukturer som eksisterer forut for språkbruk og blir gitt til språkbrukerne og 2) at språket muliggjør og begrenser språkbruk, men ikke omvendt.

Fordeler og ulemper ved en systemisk tilnærming

Selv om både Saussure og Chomsky snakker varmt om en systemisk forståelse av språk, er begge kritiske til hvorvidt en systemisk tilnærming belyser alle språklige fenomen like godt. Begge presiserer at mens språket bør forstås som et universelt og homogent system, kan konkret og individuell bruk av dette systemet fortone seg mer som en heterogen og tvetydig masse (Crowley 1990: 34f). Videre påpeker Saussure (1916: 8) at ”to distinguish between the system and its history, between what it is and what it was, seems very simple at first glance; actually the two things are so closely related that we can scarcely keep them apart”. Chomsky (1966: 37) er også skeptisk til verdien av en systemisk tilnærming til språklæring ettersom denne er både svært teoretisk og abstrakt i forhold til språkbruks varierte og konkrete sammenhenger. Slike kritiske vurderinger framhever at en systemisk forståelse er en metodisk tilnærming som kan komme i skade for å framheve noen fenomen på bekostning av andre (og ikke en sann representasjon av språklige fenomen):

”The object [of linguistics] is not given in advance of the viewpoint: far from it. Rather, one might say that it is the viewpoint adopted which creates the object. Furthermore there is nothing to tell us in advance whether one of these ways of looking at it is prior or superior to any of the others” (Saussure 1916: 8 sitert etter Crowley 1990: 27).

Når så en systemisk tilnærming brukes for å forstå språklige fenomen, rettes fokuset mot "the forces that are permanently and universally at work in all languages, and [...] the general laws to which all specific historical phenomena can be reduced" (Saussure 1916: 6). Dette perspektivet har vist seg godt egnet for å framheve mange hovedsakelig abstrakte, universelle og faste språklige fenomen, og den belyser også delvis hvordan slike strukturer muliggjør og begrenser språkbruk. Dette har gjort den systemiske tilnærmingen til språk svært produktiv, og den kan derfor regnes som et stort vitenskapelige framskritt (Pennycook 2002: 15).

Til tross for sin produktive historie, kan denne tilnærmingen likevel kritiseres for å mørklegge språklige fenomen knyttet til den mangfoldige virkeligheten ved språkbruk (Crowley 1990). Et systemisk perspektiv gjenskaper på mange måter språket som system i sitt eget bilde (jf. Saussure 1916: 8) og etablerer dermed et press om at språklige fenomen må opptre enhetlig, forutsigbart, logisk og samsvarende. Men ved å søke "unity in diversity" (Bakhtin 1981: 274, 270-5) legger den systemiske tilnærmingen lokk over hvordan andre situasjonelle, sosiale eller kulturelle krefter påvirker språkbruk. Svakheten ved den systemiske tilnærmingen ligger dermed i dens forutsetninger: "The model does crucially claim that there are universal laws of some kind, but, as a model does not specify what kind of laws they are" (Hopper 1993: 34). Tilnærmingen forutsetter altså at språket er et universelt, homogent system som eksisterer forut for språkbruk, og den ser dermed bort ifra hvordan språkbruk påvirker språkssystemet (et unntak er grammatikalisering, s. 58). Dette etablerer langt på vei en ensidig relasjonen mellom språkssystemet og språkbruk, og dette skaper store problemer for studier av: 1) språkbruk, 2) relasjonen mellom språk og språkbruk og 3) språkets opprinnelse. Jeg vil nå eksemplifisere hvordan en systemisk tilnærming både belyser og mørklegger programmerings- og programmeringsspråklige fenomen.

En systemisk forståelse av programmeringsspråk

Lærebøker i programmering på universitetsnivå er ment å både speile moderne og rådende faglig innsikt på området og være sentrale bidrag i fagutdannelsen for mange programmerere. Slike lærebøker burde derfor også være gode eksempler for hvordan forståelsesmodeller og metaforer brukes om programmering.

I sin lærebok om "programming using the Java language" gir Lewis (1998) følgende beskrivelse av et program og programmeringsspråk:

"A program is written in a particular programming language that determines the words and symbols that you can use to construct a program. A programming language also defines a set of rules that determine exactly how a programmer combine the words and symbols of the language into programming statements" (ibid.: 34).

Programmeringsspråk består av eksakte regler for hvilke programytringer som er mulige og utgjør dermed et system som legger universelle premisser for alle konkrete programytringer: programmeringsspråket eksisterer forut for programmering.

Tittelen ”Java Software Solutions: Foundations of Program Design” (ibid.) reflekterer hvor grunnleggende synet på programmeringsspråket som system er i Lewis’ tilnærming. Systemet (her ”Solutions”) danner grunnlaget for programmering (her ”Program design”), og systemet presenteres som universelt, homogent og upåvirkelig gjennom hele boka. Som Saussure (1916) fester Lewis (1998) fokuset på det symbolske systemet i språket i seg selv, og bruk av dette systemet blir underordnet. Kombinert med metaforiske referanser som ”språk”, ”ord”, ”skrivning” og ”ytringer”, etablerer Lewis dermed en klar parallell til det universelle, homogene språkssystemet.

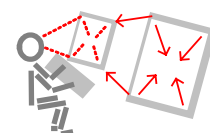
Ettersom en slik systemisk forståelse av programmeringsspråk og slike metaforiske koblinger er så dagligdags, kan det være vanskelig å påpeke hull i Lewis tilnærming. Men Lewis’ systemiske tilnærming er ikke ”sann”. Lewis belyser bare fenomen på en måte vi kjenner igjen samtidig som han mørklegger fenomenen vi ikke er vant med å få belyst. Denne umiddelbare gjenkjennelsesverdien gjør det vanskelig å skille mellom denne tilnærmingen og de bakenforliggende fenomenene. For å bedre forstå disse fenomenene, trenger vi derfor å synliggjøre dette skillet mellom tilnærming og fenomen. Jeg vil her derfor forsøke å trekke fram de konkrete og abstrakte fenomenene som denne tilnærmingen framhever. Deretter vil jeg forsøke å trekke fram andre fenomen jeg mener denne tilnærmingen skyver ut i skyggene.

Systemene i programmeringsspråk

I en konkret programmeringssituasjon blir programmereren umiddelbart stilt ovenfor et teknisk utviklingsmiljø. Dette utviklingsmiljøet består av en gruppe verktøy som får programkoden til å fungere og som programmereren kan utforme programkoden med. Slike utviklingsmiljø kan settes sammen på mange forskjellige måter med varierende kompleksitet og form, men felles for alle utviklingsmiljø er at de *direkte* og *konkret* muliggjør og begrenser hvilken programkode som vil og ikke vil fungere innenfor sine rammer. Utviklingsmiljøet danner dermed et speilbilde av reglene for hva som kan og ikke kan skrives med programmeringsspråket, og kan forstås som konkrete programmeringsspråklige system som blir *gitt til* den enkelte programmereren og som håndhever reglene i programmeringsspråket.

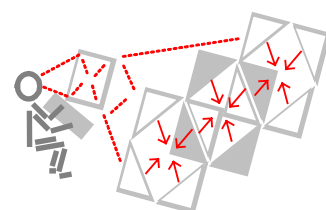


For at et program skal fungere, må det være kompatibelt med de systemenhetene som det skal samarbeide med. Utviklingsmiljøet som får programkoden til å fungere, står derfor ikke fritt til å utforme retningslinjene for programmering. Drivkraften som former mulighetene og begrensningene i utviklingsmiljøet er viljen til å få programmerens kode til å fungere sammen med andre system. Bak

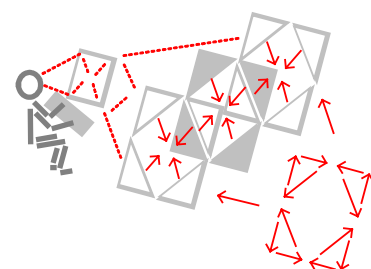


utviklingsmiljøet ligger derfor en konkret systemisk virkelighet som legger grunnlaget for de systemiske mulighetene utviklingsmiljøet kan og ikke kan presentere for programmereren. Programmereren påvirkes dermed både *direkte* av utviklingsmiljøet og *indirekte* av de konkrete systemenhetene som utviklingsmiljøet formidler. Programmeringsspråket er dermed både determinert av det konkrete utviklingsmiljøet og den bakenforliggende systemiske virkeligheten.

I praksis fungerer de konkrete systemiske virkelighetene bak utviklingsmiljøet som et kollektiv av forskjellige program og system som samarbeider. Disse kollektivene er både mangfoldige og samtidig svært avhengige av hverandre, og for at de skal kunne fungere sammen, trenger derfor de forskjellige systemene å tilpasse seg hverandre. Dette gjør at systemene blir svært tett sammenvevd og at funksjoner og regler i et system kommer i kontakt med og påvirker svært mange andre system. F. eks. kan bestemte retningslinjer for funksjonalitet i et operativsystem eller en nettleser direkte påvirke og ramme inn svært mange andre samarbeidende program. Slike retningslinjer spres også indirekte gjennom systemkollektivene som lange kjeder av samarbeidspremisser. F. eks. vil elektromekanikken i prosessorer og kabler indirekte være med å muliggjøre og begrense funksjonalitet i andre program via f. eks. et operativsystem.



Slike relasjoner og kjeder av funksjonelle retningslinjer går på kryss og tvers av de konkrete systemkollektivene. Noen av disse retningslinjene er i konflikt med andre retningslinjer på andre områder og på den måten knyttet til et bestemt system eller systemkontekst. Slike kontekstbundne eller applikasjonsspesifikke retningslinjer er heterogene sett i forhold til systemkollektivene som helhet. Men noen retningslinjer går igjen i svært store deler av den konkrete systemvirkeligheten. Slike retningslinjer som f. eks. skillete operator/operand, bruk av noder/pekere, elementær logikk og Turingmaskinen, er felles for så mange system at de kan forstås som universelle for alle system i dag. Den konkrete systemiske virkeligheten består derfor av *både* heterogene, applikasjonsspesifikke og homogene, universelle retningslinjer.



Bak de konkrete, mangfoldige systemiske virkelighetene kan det dermed skimtes konturene av et abstrakt, universelt og homogent system. Dette universelle, homogene systemet kan spores i alle konkrete program og system, og utviklingsmiljøet vil dermed også formidle dette abstrakte systemets retningslinjer til programmereren. Mulighetene og begrensningene i programmeringsspråket blir dermed også forankret i dette abstrakte systemet, vel å merke gjennom en konkret, heterogen systemisk virkelighet.

Programmeringsspråket kan dermed på et overordnet nivå forstås som rammet inn av et abstrakt, universelt og homogent "system bak systemene". Utviklingsmiljøet etablerer både heterogene og homogene rammer for hva programmereren kan og ikke kan få til å fungere, men alle heterogene retningslinjer må samsvare med noen bestemte homogene retningslinjer. Heterogene system underordnes dermed et universelt system, og det universelle systemet kan dermed forstås som gitt til programmereren og som forut for programmering. En slik forståelse av programmering og programmeringsspråk er systemisk.

2.3 Skygger fra en systemisk tilnærming til programmering

En systemisk tilnærming fungerer godt for å framheve mer eller mindre faste, universelle og homogene krefter i programmeringsspråk, men som den beslektede tilnærmingen til språk, er det flere fenomen som denne tilnærmingen ikke belyser. For det første er denne tilnærmingen svært vag om hvor de universelle kreftene bak programmeringsspråket kommer fra. Tilnærmingen fungerer godt for å sette ord på de interne strukturene i programmeringsspråket, men den redegjør ikke for hvilke krefter som står bak disse strukturene igjen og hvordan programmeringsspråk oppstår. For det andre mangler tilnærmingen en beskrivelse av hvordan programmereren påvirker programmeringsspråket. Tilnærmingen er tydelig om hvilke systemiske muligheter og begrensninger programmereren må forholde seg til, men programmereren etablerer ingen ytre mulighet og begrensning for programmeringsspråket utover de indre mulighetene og begrensningene programmeringsspråket selv etablerer. For det tredje beskriver ikke denne tilnærmingen hvilke andre krefter som også påvirker programmering og skaper heterogene krefter i systemene. Alle muligheter og begrensninger som presenteres for programmereren sees som del av eller i sammenheng med de systemiske fenomenene i programmeringsspråket, og det er ingen andre krefter utenfor dette systemet som tilnærmingen fanger inn og setter ord på.

Som jeg vil vise, kan alle disse skyggeområdene forstås som beslektet. Når jeg nå skal belyse disse skjulte fenomenene, vil jeg derfor ta utgangspunkt i opphavet til programmeringsspråk og framheve et alternativt syn på "kreftene bak programmeringsspråk". Etersom programmeringsspråk oppfattes som et teknisk fenomen, er det nærliggende å forstå systemet bak programmeringsspråk som et matematisk system. Som innfallsvinkel til kreftene bak programmeringsspråk, vil jeg derfor forsøke å skildre kreftene bak matematiske system.

Kreftene bak matematiske system

Det er én begivenhet i matematikkens historie som mer enn noen annen har kastet lys over kreftene bak matematiske system: utviklingen av ikke-euklidisk geometri. Fram mot slutten av det 19. århundre var Euklidisk geometri ansett som en abstrakt, absolutt,

universell og ”sann” representasjon av virkeligheten. Euklidsk geometri underbygget Newtons naturlover, og bl. a. Kant regnet disse logiske prinsippene som en absolutt dimensjon ved vår erkjennelse av verden. Dette systemet dannet dermed en overordnet ramme som eksisterte forut for og uavhengig av menneskelige aktiviteter (Greenberg 1974, Kline 1972: 862).

Til tross for den Euklidske geometriens intuitive samsvar med vår erkjennelse av virkeligheten, oppdaget matematikere observerbare fenomen som ikke ubetinget stemte overens med dette absolutte systemet. Dette førte til et omfattende arbeid med å utbedre betingelsene og forbedre den Euklidske geometrien, og det var paradoksalt nok dette arbeidet som førte til at alternative geometriske konstellasjoner ble utviklet. Disse geometriene var grunnleggende forskjellig fra den Euklidske og fikk derfor navnet ikke-euklidske geometrier (Greenberg 1974: 150, Kline 1972: 862).

Betydningen av ikke-euklidske geometrier og det radikale bruddet med den etablerte forståelsen av at det fantes bare ett universelt og homogent system bak matematiske og fysiske lover, kan vanskelig overdrives. Gauss (1829, her sitert etter Greenberg 1974: 146), sin tids kanskje mest anerkjente matematiker, våget f. eks. ikke å offentliggjøre denne ”revolusjonære oppdagelsen” av frykt for vitenskapelig sjikane og ”the howl from [followers of Kant]”. Ikke-euklidsk geometri åpnet nemlig for muligheten for å etablere flere indre homogene (her: samsvarende, motsigelsesfrie) logiske system som stod i et ytre heterogent motsetningsforhold til andre logiske system. En erkjennelse av eksistensen av slike heterogene system hadde den dramatiske konsekvensen at vår forståelse av verden ikke lenger kunne forstås som svart/hvit, sann/falsk, men snarere som en av mange mulige. Med ett og samme pennestrøk hadde dermed ikke-euklidsk geometri brutt med både det etablerte newtonske verdensbilde og den kantianske erkjennelsen av erkjennelsen.

Konsekvenser av heterogene geometriske system

Erkjennelsen av at kreftene bak matematikken ikke lenger kunne forstås som universelle og homogene, åpner også for en annen viktig erkjennelse. Logikken som hittil hadde vært absolutt, endelig og *gitt til* mennesket, blir plutselig levende, foranderlig og en av mange mulige, heterogene logikker *skapt av* mennesket:

”The geometrical axioms are therefore neither synthetic a priori intuitions nor experimental facts. They are conventions. [...] One geometry cannot be more true than another: it can only be more convenient.” (Poincare 1902, sitert etter Greenberg 1974: 250)

Dette snur opp ned på den etablerte forutsetningen om at relasjonen mellom matematikken og matematikeren var ensidig og absolutt. Matematikeren, ledet av sin situasjonelle erfaring og situerte valg, blir her den fremste skapende kraften bak

matematiske system. Matematikeren frigjøres fra matematikken, og det eksisterte nå et gjensidig påvirkningsforhold mellom dem.

Den individuelle, situerte matematikeren kan heller ikke forstås som den eneste skapende kraften bak matematiske system.

”The constructions of the mathematical mind are at the same time free and necessary. The individual mathematician feels free to define his notions and to set up his axioms as he pleases. But the question is, will he get his fellow mathematicians interested in the constructs of his imagination? We cannot help feeling that certain mathematical structures which have evolved through the combined efforts of the mathematical community bear the stamp of necessity not affected by the accidents of their historical birth” (Weyl, her sitert etter Greenberg 1974: 253).

Matematikeren er fri, men denne friheten er også begrenset av hans relasjon til andre matematikere og andre matematikers anstrengelser. Matematikeren operer dermed også i en sosial, historisk og matematisk omgivelse som sammen med hans individuelt situerte valg bidrar til å forme hans matematiske konstruksjoner.

Utviklingen av ikke-euklidisk geometri bryter dermed med en systemisk tilnærming til matematikk. Ettersom individuelt, sosialt, historisk og matematisk situert bruk av matematikk kan forstås som underliggende selv de mest abstrakte, fundamentale kreftene i et matematisk system, vil ikke en tilnærming som tar utgangspunkt i at matematikk er et system som eksisterer forut for slik situert bruk, fange inn disse kreftene. En systemisk tilnærming vil dermed mørklegge 1) kreftene bak matematikken, 2) den gjensidige dialektikken mellom matematikeren og matematikken og 3) situasjonell, sosial og historisk situert bruk av matematikk.

Kreftene bak programmeringsspråk

Med utgangspunkt i at overordnede, abstrakte og universelle regler for programmering kan forstås som et matematisk system, kan dermed også programmeringsspråkets opphav forstås som definert og *skapt* av programmerere som svar på skiftende situasjonelle, sosiale og historiske behov. En slik påvirkning *fra* programmerer *til* teknologi undergraver ikke en påvirkning *fra* teknologi *til* programmering. Disse motsatte kreftene kan eksistere samtidig, og dermed kan programmering og programmeringsspråk forstås som å *gjensidig* etablere muligheter og begrensninger for hverandre.

En gjensidig forståelse av relasjonen mellom programmering og programmeringsspråk sidestiller ikke bare programmereren med de tekniske systemene. En slik tilnærming synliggjør også at programmering påvirkes av krefter utenfor teknologien og programmererens kreativitet; også situasjonelle, sosiale og

historiske forhold etablerer muligheter og begrensninger for programmereren. Slik ytre, ikke-teknisk påvirkning er velkjente krefter for programmerere, men disse ikke-systemiske kreftene behøver ikke lenger å forstås som ensidig underordnet teknologien. Situasjonelle, sosiale og historiske krefter kan også forstås som gjensidig formende for programmering og dermed teknologien. Programmering og programmeringsspråk blir dermed gjensidige prosesser mellom programmerere, teknologi, programmering og situasjonelle, sosiale og historiske omgivelser.

Behovet for en ny tilnærming til programmering

Dette gjensidige samspillet bryter med en systemisk tilnærming. En systemmetafor gir riktignok assosiasjoner mot samspill, men disse går i retning av et indre samspill og relasjoner mellom deler i én statisk enhet. Én overordnet enhet kan derimot vanskelig omslutte påvirkning fra programmereren eller situasjonelle, sosiale og historiske omgivelser på programmeringsspråket og det ytre, levende og foranderlige samspillet i disse prosessene. 1) Opphavet til programmeringsspråk, 2) relasjonen mellom programmering og programmeringsspråk og 3) hvordan situasjonelle, sosiale og historiske krefter påvirker programmering vil derfor falle utenfor fokuset til en systemisk tilnærming som forstår programmeringsspråket som *gitt til* programmereren *forut for* programmering.

For å belyse disse tre fenomenene, trenger vi derfor å gå bort ifra de systemiske forutsetningene og åpne våre øyne for andre forutsetninger. Vi trenger en tilnærming som 1) åpner for en gjensidig relasjonen mellom programmering og programmeringsspråk og 2) fanger inn språkbruk i dens konkrete, situasjonelle og sosiale omgivelser. En slik ikke-systemisk tilnærming vil bedre kunne synliggjøre og konkretisere disse tidligere lysskye fenomenene som også står sentralt i utvikling av multiagentsystem og derfor er en forutsetning i dette prosjektet (jf. s. 1-2).

2.4 En dialogisk tilnærming til språk

I en dialogisk tilnærming er ”dialog” den bærende metaforen og forståelsesmodellen. I motsetning til systemmetaforen som retter tankene mot interne relasjoner *i* språk, retter dialogmetaforen tankene mot språkbruk og samspillet *mellom* individuelle språkbrukere og *mellom* språkbrukere og språk. Språkbruk forstås som den primære kraften bak både språk, språkbrukere og deres omgivelser, og det metodiske fokuset rettes derfor mot hvordan disse prosessene gjensidig former hverandre (Evensen 2002b, Holquist 1990). Dette står i sterk kontrast til den systemiske tilnærmingen som forstår språkbruk som sprunget ut av språksystemets og individets skapende krefter, og derfor fokuserer på hvordan språk og individ former språkbruk.

Min dialogiske tilnærming baserer seg på Evensens (2002b) beskrivelse av språkbruk. Innenfor en slik tilnærming kan i utgangspunktet alt forstås som dialogisk (Holquist

1990), og dette brede fokuset gjør denne tilnærmingen anvendelig på mange forskjellige problem. Men den filosofiske dybden som følger med et slikt bredt fokus, gjør også begrepene dialog og språk svært abstrakte. Det er derfor nødvendig å 1) begrense omfanget av hvilke dialogiske prosesser jeg vil undersøke og 2) utdype og konkretisere forståelsen av disse dialogiske og språklige prosessene.

Med denne tilnærmingen vil jeg i neste kapittel først begrense meg til den gjensidig skapende dialogen mellom språk og språkbrukere. Denne dialogen vil jeg utdype med ved å hente fram og presentere noen av Bakhtins metaforer (1981, 1984, 1998) for ”stemme”, ”heteroglossia”, ”polyfoni” og ”refraktering”. Deretter vil jeg utvide omfanget av denne dialogen og vise hvordan situasjonelle og sosiale krefter også påvirker språkbruk. Denne andre dialogiske dimensjonen vil jeg utdype med metaforen ”dobbel dialog”.

Språk som ytringer, stemmer og heteroglossia

Når språk forstås som konstituert av gjensidige prosesser mellom språket selv og språkbrukere, framheves konkret språkbruk som den skapende kraften bak språk. Konkret språkbruk blir dermed objektet som skal undersøkes og trenger derfor også å kategoriseres. For å gjøre dette, bruker den dialogiske tilnærmingen begrepene *ytring* og *stemme* som rotkategorier.

”An utterance, spoken or written, is always expressed from a point of view [a voice]” (Clark & Holquist sitert etter Wertsch 1991: 51). Wertsch (ibid.) beskriver språkbruk som bygget opp av konkrete ytringer som i sin tur uttrykker et perspektiv og en stemme. Men selv om en ytring kan forstås som å uttrykke en stemme, kan språket ytringen er formet i samtidig forstås å uttrykke flere stemmer:

”[Language] is populated - overpopulated - with the intentions of others. [...] The living utterance, having taken meaning and shape at a particular historical moment in a socially specific environment, cannot fail to brush up against thousands of living dialogical threads” (Bakhtin 1981: 294, 276).

En ytring kan dermed forstås som både en konkret stemme og et abstrakt stemmepotensial.

Gjennom sin kontakt med andre ”levende dialogiske tråder”, etablerer ytringene en bro *fra* den konkrete stemmen i den konkrete språkbrukssituasjonen *til* stemmer fra andre tidligere, samtidige eller fremtidige språkbrukssituasjoner. Arenaen for denne kontakten er ytrers og tilhørers bevissthet, og broen (dialogen) mellom stemmene i språket skapes ved en aktiv fortolkning og konnotasjon i språkbrukernes perspektiv: ”Actual meaning is understood against the background of other concrete utterances on the same theme, a background made up of contradictory opinions, points of view and

value judgements” (ibid.: 281). Det blir dermed språkbrukernes assosiasjoner til andre konkrete stemmer som skaper de interne relasjonene mellom stemmer, og ytringer og stemmer kan derfor samtidig eksistere i samsvar og motsvar med hverandre. (Senere vil jeg komme tilbake til bl. a. adressivitet og andre egenskaper ved ytringer og stemmer.)

Språkbruk er prosessene der språklige stemmer både skapes og gjenskapes i språkbrukernes bevissthet. Disse stemmene lever og er i utvikling fordi språkbrukerne hele tiden puster liv i gamle stemmer samtidig som de skaper nye stemmer. Språket kan dermed forstås som bygget opp av ytringer og stemmer som hele tiden fornyer seg og tar form som et stemmelandskap der ytre og indre grenser ikke er entydig fastsatt. Språket skapes dermed ”in the process of living interaction” som et mangfoldig, samsvarende og motsvarende landskap, et *heteroglossia* (ibid.: 282). (Se (ibid.: 270ff) for mer om heteroglossia og sentrifugale og sentripetale krefter.)

Språkbruk som polyfoni og refraktering

Over har jeg beskrevet hvordan konkrete stemmer bygger opp en heteroglossia og påvirkningen *fra* språkbruker *til* språk. Men denne påvirkningen går også motsatt vei, og språket kan også forstås som formende for språkbrukeren.

”The author does not speak in a given language (from which he distances himself to a greater or lesser degree), but he speaks as it were, through language, a language that has somehow more or less materialized, become objectivized, that he merely ventriloquates. [...] He] makes use of words that are already populated with the social intentions of others and compels them to serve his own new intentions, to serve a second master. Therefore [the author’s intentions] are refracted, and refracted at different angles. [...] Diversity of voices and heteroglossia enter the novel and organize themselves within it into a structured artistic system” (Bakhtin 1981: 299f).

Språkbruk er å skape egne stemmer og intensjoner ved gjenbruk av andres stemmer og intensjoner: som omvendte lysstråler i en prisme, skapes fargerike stemmer gjennom refraktering og sammensmelting av andre fargerike stemmer.

En annen måte å forstå språkbruk som gjenbruk av stemmer på, er som en ”polyfoni”. Når språkbrukeren konstruerer sin egen stemme, gjør han dette ved å orkestrere stemmer i en symfonisk ytring, hvis essens ”lies precisely in the fact that the voices remain independent and, as such, are combined in a unity of a higher order than in homophony” (Bakhtin 1984: 21). Språkbruk og konstruksjon av nye stemmer tar således form som en kunstnerisk gjenbruk og sammenblanding av andre selvstendige stemmer. Denne sammenblandingen er ”a plurality of independent and unmerged voices and consciousnesses, a genuine polyphony” (ibid.: 6), og den polyfoniske ytringen kan derfor både være i harmoni og disharmoni samtidig.

Språkbruk som dobbel dialog

Beskrivelsene over presenterer språkbruk som en dialog mellom språkbruker og språk. Men denne dialogen er ikke det eneste gjensidige samspillet i språkbruk. I en konkret, sosial situasjon påvirker også språkbrukere hverandre og sine omgivelser gjensidig. Denne konkrete dialogen mellom språkbrukere kan forstås som:

”The specific, symbolic meeting place between a specific, situated writer and an equally specific, but differently situated reader(s) (Nystrand 1986, 1989). At this meeting point, meaning is negotiated across differences in situatedness, knowledge and perspectives” (Evensen 2002b: 383).

Språkbruk kan derfor også forstås som konkret forhandling om mening mellom situerte språkbrukere med hver sine perspektiv. I disse forhandlingene søker språkbrukere å utvikle en rimelig grad av enighet i sine syn på virkeligheten, ”a temporarily shared social reality” (Rommetveit sitert etter Evensen 2002b: 399). Dette gjør de gjennom gjensidig påvirkning og tilpasning, en slags ”attunement to the attunement of the other” (ibid.).

Den gjensidige påvirkningen mellom språkbrukernes perspektiv stikker dypere enn bare en gjensidig tilpasning. ”Å bestemme sin egen posisjon utan å setje han i samband med andre posisjonar er umogleg” (Bakhtin 1998: 35). Språkbrukerne er dermed avhengige av hverandres perspektiv for å konstruere sine egne, og utformer sine stemmer som proaktive responser til hverandre. ”Ytringar er ikkje likegyldige overfor kvarandre og er ikkje seg sjølv nok, dei veit om kvarandre og reflekterer kvarandre gjensidig” (ibid.). Ytringene danner en bro mellom språkbrukeres konkrete perspektiv, og perspektiv og stemmer mot- og samsvarer alltid med andre perspektiv og stemmer.

Språkbrukernes konkrete perspektiv er således betinget av andre språkbrukeres perspektiv, og språkbrukerne er på denne måten forankret i en sosial omgivelse av andre, deres ytringer og perspektiv. ”Det essensielle (konstitutive) kjennemerket til ytringa er at ho blir *retta til nokon*, - ytringas *adressivitet*” (ibid.: 39). Men denne intertekstuelle omgivelsen av ytringer, stemmer og perspektiv er ikke bare sosial. ”Ytringa representerer [også] talarens aktive posisjon i ei eller anna tematisk sfære” (ibid.: 27). Videre etablerer ytringa språkbrukerens situasjonelle posisjon og perspektiv: ”Everything is perceived from a unique position in existence” (Holquist 1990: 21).

”At the very basic level, then, dialogism is the name not just for a dualism, but for a necessary *multiplicity* in human perception” (Holquist 1990: 22). En språkbruker vil altså etablere et eget perspektiv situert i en situasjonell, tematisk, sosial og intertekstuell omgivelse. Hver stemme vil speile dette perspektivet i sin relasjon til

andre stemmer, og gjennom å *orkestrere* andres sosialt og situasjonelt forankrede perspektiv og stemmer, vil språkbrukerne dermed samtidig skape sin egen situasjonelle, sosiale og tematiske posisjon: "Thought is not merely expressed in words; it comes into existence through them" (Vygotsky 1997: 218). Språkbruk blir dermed en prosess der språkbrukere utformer sine tanker og perspektiv, og "det er berre [denne prosessen] mellom språkleg betydning og den konkrete røynda, berre kontakten mellom språket og røyndommen som finn stad i ytringa, som tenner ekspressivitetens gneiste: han førekjem korkje i språksystemet eller i den objektive røyndomen som omgjev oss" (Bakhtin 1998: 30). Slik skaper språkbrukerne sin "midlertidige, sosiale og delte virkelighet" og påvirker dermed sine egne og hverandres omgivelser gjennom språkbruk. Stemmene og den polyfoniske stemmebruken formgis i en konkret sosial dialog mellom språkbrukere, og denne konkrete dialogen henter samtidig de muliggjørende formene i "an available [heteroglot] resource pool for immediate communicative needs and pressures" (Evensen 2002b: 404).

Påvirkningen på den enkelte språkbrukers perspektiv skjer ikke bare fra språket eller andre språkbrukere og fra de situasjonelle omgivelsene. Påvirkningen på språkbrukeren kommer fra språkbruksprosessen som helhet. Det totale bildet av språkbruk blir dermed som en dobbel dialog, dvs. en umiddelbar, gjensidig dialog mellom situerte språkbrukere som i sin tur også er i dialog med dialogen mellom språkbrukerne og språket (Evensen 2002b: 385). En tilnærming til språk og språkbruk som ønsker å fange opp denne doble dialogen, må derfor ta utgangspunkt i denne helheten.

Stemmer som sosiale skapte ressurser: grammatikalisering

I denne oppgaven legger jeg spesiell vekt på hvordan situasjonelle og sosiale krefter påvirker språkbruk og hvordan denne påvirkningen over tid også påvirker språket. Dette har to årsaker. For det første har situasjonelle og sosiale krefters påvirkning på språket en klar nytteverdi i prosjektet. Vårt arbeid var og er begrunnet i situasjonelle og sosiale programmeringsbehov, og dette arbeidet resulterte i utviklingen av nye programspråklige ressurser. Det vil derfor være nyttig å etablere en parallell til språkvitenskapelige studier som belyser denne utviklingsprosessen. For det andre har en slik påvirkning klare implikasjoner for gyldigheten av en systemisk forståelse av programmeringsspråk. En grundigere forståelse av hvordan språkbruk og eventuelt situasjonelle og sosiale krefter påvirker språket vil synliggjøre begrensningene i en systemisk tilnærming og dermed også underbygge behovet for min dialogiske tilnærming.

Evensen (2002b) synliggjør hvordan sosiale og situasjonelle krefter påvirker språkbrukeren til å gjenbruke egne og andres stemmer. Gjennom gjenbruk sprer disse

konkrete stemmene seg mellom forskjellige sosiale og kulturelle grupperinger, og abstraheres gradvis til mer generelle språklige mønstre og sosialt gjenkjennelige konvensjoner. Miller (1984: 31) underbygger denne forståelsen av språklige konvensjoner i sin gjennomgang av ”genres as typified rhetorical actions based in recurrent situations”. Situasjonelle og sosiale krefter følger et gjentakende kommunikativt mønster som gjør at de etter hvert sedimenteres til språklige konvensjoner som språkbrukere i sin tur kan gjenbruke og skape mening med.

Slike sosialt forankrede studier av utvikling av språklige konvensjoner er interessante, men til tross for dette, vil jeg her bare fokusere på studier av hvordan grammatiske strukturer formes over tid, grammatikalisering. Dette valget har jeg tatt fordi disse studiene har klare og oppsiktsvekkende paralleller til utviklingen av (pro)grammatiske språklige strukturer i vårt prosjekt.

Grammatikalisering brukes om ”the processes whereby lexical items and constructions come in certain linguistic contexts to serve grammatical functions, and, once grammaticalized, continue to develop new grammatical functions” (Hopper 1993: xv). Eksempler på slike ”historiske overganger fra mer leksikalske til mer grammatiske kategorier [er] når substantivet bak blir til preposisjon, eller når det selvstendige demonstrativet *inn* blir til den etterhengte artikkelen *-en* i skandinavisk” (Faarlund 2002: 61, min utheving). *Bak* var f. eks. bare en referanse til rompa til å begynne med. Men denne referansen kunne også brukes for å metaforisk beskrive en mer generell romlig relasjon mellom rompa og forsiden av kroppen. Etter hvert som dette metaforiske bildet ble mer og mer brukt, ble *bak*'ens kroppslige konnotasjon utvasket og den mer generelle, romlige relasjonen framhevet. En lignende historie har også den morfologiske bøyningen som skiller bestemte fra ubestemte substantiv i norsk. Desto oftere ordet *inn* ble brukt rett etter substantivet for å påpeke at det var et bestemt objekt det var tale om, desto mer oppfattet språkbrukerne en sammenstilling av de to ordene substantiv+*inn* som én enhet. Til slutt ble denne assosiasjonen så vanlig at *inn* ble til en morfologisk bøyning av substantivet: *-en* (jf. Hopper 1993: 63-93).

På denne måte utvikles selv de mest grunnleggende av språkets grammatiske strukturer: ”The synchronic facts [, syntax and morphology,] were indistinguishable from the diachronic and discourse pragmatic process they were caught up in” (Hopper 1993: 28). Det er imidlertid strid om hvorvidt denne utviklingen kan forstås som et internt lukket system eller et åpent system drevet av situasjonelle og sosiale krefter. Hopper (ibid.: 67) ser språket som et enhetlig lukket system ettersom ”communicative ‘need’ is not a plausible motivation in most cases of grammaticalization [and] violates the uniformitarian principle”. Givon (sitert etter Hopper 1993: 29) mener derimot at disse prosessene bør forstås som ytre, situasjonelt og sosialt drevet og at lingvistiske

regler og kategorier er grunnleggende avhengige av sin funksjon. Denne faglige uenigheten markerer et veiskille mellom en systemisk og en dialogisk tilnærming: Et syn om at utvikling av grammatiske strukturer er et lukket system ensidig sprunget ut av f. eks. generelle kognitive prosesser (ibid.), er en systemisk forståelse av språkhistorie; et syn om at utviklingen av grammatiske strukturer er en åpen prosess der situasjonelle, sosiale krefter gjensidig påvirker språket, er derimot en dialogisk forståelse av språkhistorie.

I en dialogisk tilnærming til språkhistorie har grammatiske strukturer sitt opphav i helt konkrete ytringer og stemmer: "Yesterday's pragmatics is today's syntax [is tomorrow's morphology]" (Givon sitert etter Faarlund 2002: 62). Disse konkrete ytringene og stemmene blir formet gjennom en gradvis framheving av generelle stemmeaspekt og en gjentagende bruk av ytringsaspekt som når språkbrukerne møter ordet *bak* i konkrete ytringer og gradvis framhever den uskyldige og romlige stemmen og gradvis overser den grove og kroppslige stemmen. Slik påvirker språkbrukerne og deres situasjonelle og sosiale omgivelser gradvis leksikalsk bærede stemmer til å skifte form og opptre som grammatiske stemmer. Grammatiske strukturer blir dermed språklige konstruksjoner som bærer stemmer i språkbrukernes aktivt fortolkende bevissthet.

Fordeler og ulemper ved en dialogisk tilnærming

I forhold til språk og språkbruk skiller den dialogiske tilnærmingen seg fra den systemiske tilnærmingen. Dette gir både fordeler og ulemper, og jeg vil begynne med ulempene og vise hvordan disse kan snus til forskerens fordel.

Saussure (1916: 9) peker på at en tilnærming som ønsker å favne om språkbruk i alt sitt mangfold vil bli svært omfattende: "If we study speech from several viewpoints simultaneously, the object of linguistics appears to us as a confused mass of heterogeneous and unrelated things." De gjensidig, mangfoldige prosessene kan dermed bli en ond sirkel som forskeren aldri får riktig grep om. Dette vil gjøre det svært vanskelig å skape både den oversikten og dybdeforståelsen som forskeren trenger for å utvikle konkrete og relativt stabile beskrivelser av språkbruk, språket og språkbrukeren.

Men Saussure (ibid.) peker videre på at "if we fix our attention on only one side of each problem, we run the risk of failing to perceive the dualities pointed out above." Dersom vi er interessert i en forståelse av de gjensidige prosessene en dialogisk tilnærming belyser, kan vi derfor heller ikke overse behovet for et såpass omfattende faglig perspektiv. Forskeren kommer i så tilfelle ikke utenom den heterogene, sprikende massen av språkbruk, språk, språkbrukere og deres situasjonelle, sosiale og historiske omgivelser.

Den systemiske tilnærmingen ble etablert på et tidligere tidspunkt da det faglige landskapet rundt disse heterogene prosessene var langt fattigere. Evensen (2002b: 408) peker på at vi i en dialogisk tilnærming i dag kan ta med innsikt som ikke var tilgjengelig for forskere på Saussure og Bakhtins tid. I dag kan forskere basere seg på interaksjonistiske og konstruksjonistiske studier, og interaksjonisme gir forskeren dybdeforståelse i dialogen mellom situerte språkbrukere og konstruksjonisme i dialogen mellom språkbrukere og språket. ”The dialogical alternative is a genuine synthesis of interactionism and constructionism” (ibid.: 385), og sammen gjør disse studiene det mulig å danne en oversikt over de heterogene og onde språkbrukssirkelene i en dialogisk tilnærming.

Den dialogiske tilnærmingen beskriver språkbruk og gjensidige relasjoner bl. a. mellom språkbruk og dens situasjonelle, sosiale og språklige omgivelser. Denne tilnærmingen burde derfor være godt egnet for å utdype vår forståelse av relasjonen mellom programmering og programmeringsspråk, opphavet til programmeringsspråk og relasjonen mellom programmerere og andre ytre, ikke-systemiske krefter. Jeg vil nå forsøke å anvende en dialogisk tilnærming på programmering og programmeringsspråk ved å beskrive programmering som gjensidig skapende dialoger. For å gjøre dette vil jeg først og fremst fokusere på: 1) hvordan programmering kan forstås som polyfonisk orkestrering av programmeringsspråklige stemmer, 2) hvordan utvikling av programmeringsspråk kan forstås som en situasjonelt og sosialt drevet utviklingsprosess og 3) hvordan programmeringsspråk og andre program kan forstås som et heteroglossisk landskap der både tekniske, situasjonelle, sosiale og økonomiske krefter kjemper mot og med hverandre.

2.5 En dialogisk tilnærming til programmering

I dette delkapittelet vil jeg vise hvordan programmering kan forstås i henhold til en dialogisk tilnærming og dermed samtidig utdype en dialogisk forståelse til også å inkludere programmering og programmeringsspråk. Utgangspunktet for en beskrivelse av programmering som språkbruk er rotkategoriene *ytring* og *stemme*. Konkrete programytringer utgjør alt fra en enkelt programsetning til store sammensatte system. Disse programytringene settes sammen som en kombinasjon av tekniske symbol, regler og retningslinjer, og programmereren kan realisere sine programytringer både som program- og maskinvare.

Sett fra et språkvitenskapelig utgangspunkt kan de tekniske symbolene, reglene og retningslinjene forstås som leksikalske, syntaktiske og konvensjonelle ressurser. Programmereren vil bruke disse ressursene til å tolke sine problem, formulere sine løsninger og dermed danne seg et perspektiv. De tekniske ressursene og programytringen vil således mediere et konkret, programmeringsspråklig perspektiv (Ørstavik 2001).

Videre utvikler programmerere sine program for å fungere i samspill med andre programytringer i en konkret systemverden. Programmereren må relatere sine programytringer direkte i forhold til andres programmer, og dette posisjonerer også programmerernes perspektiv indirekte i forhold til hverandre. Programmering kan dermed forstås som en samtidig orkestrering av leksikalske, syntaktiske og konvensjonelle stemmer i polyfoniske programytringer som fungerer både i samsvar og motsvar med hverandre. Jeg vil nå utdype dette synet.

Bruk og utvikling av konvensjonelle stemmer i programmering

Det er et generelt problem i programmering at det finnes så ekstremt mange måter å kombinere tekniske symbol og regler på. Denne friheten åpner for å lage så særegne program at de umulig kan samarbeide med hverandre i praksis. Gjennom aktivt å posisjonere sine program med og mot andre program, kan derimot programmerere skape seg noen overordnede holdepunkter i dette potensielle mangfoldet.

Slik posisjonering gjøres ved å direkte relatere programytringer til helt konkrete programytringer i andre program. Eksempler på slike direkte relasjoner mellom program er en nettleser som baserer seg på et operativsystem eller en database som samarbeider med et nettbasert skript i bl. a. en nettbank. Men vel så viktig som direkte relasjoner mellom konkrete programytringer, er det å posisjonere programytringer og program til overordnede, generelle konvensjoner for komponering av programytringer.

Et eksempel på en slik konvensjon er skillet mellom datalager og dataprosessering som de fleste programmere etterfølger. Denne konvensjonen brukes av flere programmerere samtidig, og når programmerere velger å organisere sine programytringer etter dette overordnede prinsippet, blir det langt enklere å få deres enkelte programytringer til å fungere sammen. Dersom to programmerere f. eks. skal utvikle en nettbank, vet begge at data om transaksjoner og kontoinformasjon skal skilles fra skjermbilder og prosessering av transaksjonene. Dette skillet gir begge et felles, indirekte holdepunkt allerede før de begynner å tilpasse konkrete programytringer, og denne konvensjonen gjør nettbanken enklere å forholde seg til for begge programmererne. Ved å følge denne konvensjonen kan programmerere dermed f. eks. basere seg på og ta i bruk andre programmereres databaseløsninger og applikasjonsplattformer. Slike sosiale og kulturelle konvensjoner har derfor en sentral posisjon i de fleste moderne superkomplekse system.

For å utdype min beskrivelse av konvensjoner for sammenveving av tekniske symbol og regler, vil jeg her presentere ”design patterns”.

“One thing expert designers know not to do is solve every problem from first principles. Rather, they reuse solutions that have worked for them in the past. When they find a good solution, they use it again and again. Such experience is part of what makes them experts. Consequently, you’ll find recurring patterns of classes and communicating objects in

many object-oriented systems. These patterns solve specific design problems and make object-oriented designs more flexible, elegant, and ultimately reusable. They help designers reuse successful designs by basing new designs on prior experience. A designer who is familiar with such patterns can apply them immediately to design problems without having to rediscover them” (Gamma 1994: 1).

Design patterns er i utgangspunktet situasjonelt forankret, og programmerere utvikler selv design patterns som svar på konkrete programmeringsproblemer. Disse mønstrene utruker i sin tur programmereren med gjenbrukbare løsninger som han kan se sine konkrete problemer opp imot. Når programmerere så utveksler situasjonelt utviklede designmønstre seg imellom, blir deres konkrete løsningsperspektiv synliggjort som mulige løsninger og posisjoner. Programmererens problem, løsninger og perspektiv smelter dermed sammen i hans programytringer til konvensjoner for hvordan forskjellige situasjonelle behov og problemer kan imøtekommes. Disse konvensjonene utvikles over tid som ”en del av folkloren i det objektorienterte utviklersamfunnet eller elementer av noen vellykkede objektorienterte systemer” (ibid.: 2.), og blir bærere av sosiale, kulturelle stemmer for hvordan programmerere kan relatere sine programytringer til hverandre.

Dersom først et problem er etablert og en løsning formulert i en programytring, skaper denne programytringen både et eksempel til etterfølgelse og et mulig referansepunkt for andre programytringer. Som eksempel til etterfølgelse og referansepunkt etablerer dermed alle programytringer en mulig begynnelse av en konvensjon. Programytringer kan i praksis ikke unnlate å posisjonere seg til begynnende og videreførte konvensjoner i andre programytringer den direkte eller indirekte referer til. Dersom f. eks. en programytring ikke følger et etablert design pattern, vil den likefullt direkte eller indirekte stå i motfase i forhold til andre programytringer som følger dette mønsteret. Alle programytringer har et ståsted og et perspektiv i den konkrete programvirksomheten de realiseres i, og så lenge programmerernes stemmer er en del av en større programteknisk virkelighet, er de i dialog med hverandre.

Bruk og utvikling av leksikalske stemmer i programmering

I alle utviklingsmiljøer møter programmereren også et register av mulige symboler. Sannsynligheten av dette registeret varierer mellom alt fra to enkle heltall til et bibliotek av funksjonskall, klasser og andre ”ord”. Disse registrene utgjør på mange måter leksikalske ressurser som programmere kan sette sammen til programytringer. Når programmereren bruker symboler og ord til å bygge en konkret programytring, får disse en konkret betydning i forhold til rammene i det utviklingsmiljøet eller den konkrete systemplattformen de framføres på. Ordene og symbolenes betydning forankres således i en konkret omgivelse.

Betydningene ordene og symbolene tilordnes, skapes først og fremst av deres relasjon til andre ord og symboler internt i én programytring og eksternt i forhold til andre programytringer. Disse programytringene kan realiseres i både program- og

maskinvare, og disse relasjonene kan dermed gå på tvers av fysiske og situasjonelle grenser. Ett enkelt ord eller symbol kan samtidig ha mange interne og eksterne relasjoner i en konkret omgivelse. Disse samtidige relasjonene forsøker programmereren å veve sammen slik at de imøtekommer hans situasjonelle og sosiale behov i den konkrete programytringen. Når denne veven av indre og ytre relasjoner fungerer, vil det ”ferdige”, meningsfulle programmet fungere i et indre og ytre koherent samarbeid med andre ”ferdige” system.

Ordene og symbolene i programytringene etablerer dermed mange samtidige relasjoner og posisjoner i forhold til andre interne og eksterne programytringer. Det essensielle med ordene og symbolene er derfor ikke en abstrakt, universell betydning, men deres relasjon og posisjon til andre konkrete ord og symbol (realisert i både program- og maskinvare). Disse posisjonene er gjensidige, og ordene og symbolene blir således bærer av programmerers situasjonelle, sosiale og programmeringspråklige posisjon i forhold til andres posisjon. Leksikalske ressurser i programytringer har en adressivitet til andre programytringer, og programytringer får dermed en gjenklang av andre programmerers perspektiv. Programytringer og løsninger er således ikke bare programmerers egne, men snarere en mangfoldige posisjon satt sammen som en samling direkte adresserer til andre programmerers posisjon.

Som symbol og ord kan programmerere bruke programmeringsspråkets leksikalske ressurser til å fremme egne perspektiv *bare* i forhold til andre(s) programytringer og perspektiv. Ved å veve sammen ord og symbol i programytringer etablerer de egne løsninger på egne problem, men deres løsninger baserer seg på og inkluderer samtidig mange andre programmerers løsninger. Deres programytringer refrakterer andre programmerers programytringer for å svare på deres egne situasjonelle og sosiale behov. De leksikalske ressursene er således bærere av mulige stemmer i programmerers ytringer, og disse stemmene er byggestene for programmerers egne problem, løsninger og perspektiv.

Leksikalske ressurser i programmeringsspråk er ikke absolutte og endelige. Programmerere utformer også egne ord og symboler og redefinerer dermed de leksikalske mulighetene for deres problem og løsninger. Objektorienterte programmeringsspråk, og i særdeleshet Java (Ecker 2001: 92ff), er spesielt tilrettelagt for programmerers utvikling av nye leksikalske ressurser. Gjennom først å samle programytringer i tydelig avgrensede enheter og deretter bruke egne og andres enheter ved symbolske referanser, kan Java-programmerere dele sine forandringer og relatere sine leksikalske bærede stemmer til hverandre (Darwin 2001: 685ff, 21, jf. JCP 2003).

Java er svært åpent for programmerers påvirkning på leksikalske ressurser i programmeringsspråket. Men Java har også noen spesielle leksikalske enheter, såkalte ”reserverte ord” (bl. a. ”class, public) og symboler (bl. a. ”+”). Disse ordene og

symbolene er svært lukket for ytre påvirkning, og programmerere kan ikke endre disse uten samtidig å endre underliggende tekniske eller logiske forutsetninger i utviklingsmiljøet. Disse lukkede, leksikalske enhetene er på mange måter markører for syntaktiske og morfologiske strukturer, og endringer av disse leksikalske enhetene kan derfor forstås som endring av grammatiske ressurser i programmeringsspråket. Jeg vil til slutt derfor diskutere grammatiske stemmer i programmeringsspråk og den gjensidige påvirkningen mellom programmereren og disse stemmene.

Bruk og utvikling av grammatiske stemmer i programmering

I utviklingsmiljøet møter programmereren logiske regler for hvordan ord og symbol kan settes sammen i fungerende programytringer. Slike grammatiske reglene fungerer samtidig med de leksikalske og konvensjonelle ressursene og er nødvendige for å skape relasjoner mellom indre og ytre programytringer. Sett fra programmererens ståsted kan de grammatiske reglene bak programytringen virke underforståtte, abstrakte og endelige, men også disse reglene har en konkret og foranderlig forankring. F. eks. kan løkkestrukturer oversettes til en serie ”jump”-kommandoer for minneforflytninger, og selv skillet mellom operator og operand i maskinkode er en eksplisitt kommando for instruksjonsdekoderen i prosessoren. Ved endringer i disse grunnleggende strukturene vil også de logiske programmeringsspråklige reglene endres. De grammatiske reglene i en programytring skaper dermed også konkret relasjoner til andre programytringer i både program- og maskinvare. Og ettersom programmereren svært ofte er nødt til å forholde seg til flere slike regler samtidig, øver dette settet av logiske, grammatiske regler stor påvirkning på programmereren.

Det er en kjensgjerning at programmeringsspråk er i utvikling (Holmevik 1995), men en systemisk tilnærming har likevel en tendens til å oppfatte de logiske reglene som en bestandig, universell og homogen kjerne i programmeringsspråk (Lewis 1998: 39). Utformingen av utviklingsmiljø støtter derfor sjelden påvirkning fra programmereren på de logiske reglene i programmeringsspråk.⁷ Troen på en bestandig logisk grammatikk i programmeringsspråk fører derfor til at selv enkle grammatiske endringer avstedkommer et ”nytt språk” snarere enn ”programmeringsspråklige dialekter”.

Et eksempel på slik påvirkning fra programmerere på grammatiske ressurser i programmeringsspråk er utviklingen av programmeringsspråket SIMULA og en logisk regel for å lage objekter (Holmevik 1995). Utviklingen av SIMULA og objekter som syntaktisk regel var i utgangspunktet drevet av konkrete problem:

⁷ Det finnes unntak fra denne regelen (jf. ”precompiling”), men grunnet dagens oppbygning av særlig kompilator og run-time-miljø, krever endringer av grammatiske strukturer stor innsikt i programmering og utviklingsmiljøet og svært omfattende prosesser (jf. JCP 2003).

“For operational research workers and others studying network systems, there is a definite need for a precise language which allows a description of a network in terms of standardized and generally accepted concepts. [...] SIMULA ‘SIMULATION LANGUAGE’ represents an effort to meet this need with regard to discrete event-networks, that is, where the flow may be thought of as being composed of discrete units demanding services at discrete service elements, and entering and leaving the elements at definite moments of time.” (Nygård sitert etter Holmevik 1995).

Med utgangspunkt i egne problem og behov designet programmererne Dahl og Nygård SIMULA som ”en simuleringsfunksjonspakke sammen med en preprosessor til ALGOL 60”. Bakgrunnen for å bygge videre på programmeringsspråket ALGOL60 var at andre programmerere lettere skulle kunne relatere seg til programytringene i SIMULA og at programytringer utviklet med SIMULA, i sin tur skulle være kompatible med eksisterende program- og maskinvare. En direkte relasjon til ALGOL60 ville imøtekomme disse behovene. Etter hvert viste det seg likevel at noen logiske, grammatiske regler i ALGOL60 vanskelig lot seg forene med de logiske reglene Nygård og Dahl ønsket å etablere i SIMULA: ”ALGOL60’s procedure calls and storage allocation mechanisms operated strictly according to a stack principle, whereas objects [...] in a simulation modell rather tended to behave according to the que principle”. Denne konflikten førte til at programytringene og de grammatiske reglene i SIMULAs funksjonspakke og preprosessor gradvis ble løsrevet fra relasjonen til ALGOL60 og begynte å relatere seg direkte til flere andre program og system som lå nærmere maskinvaren (ibid.).

Programmererne bak SIMULA ble således både drevet av konkrete problem og behov samtidig som de ble påvirket av sine sosiale, kulturelle og programmeringsspråklige omgivelser. I en periode på omtrent fem år med kontinuerlig forhandling med disse omgivelsene utviklet de gradvis programytringer som resulterte i en logisk, grammatisk regel for bruk av objekter som syntaktisk programmeringsressurs. Denne syntaktiske regelen var hele tiden forankret i konkrete programytringer i bl. a. preprosessoren og de stod hele tiden i direkte og indirekte relasjon og posisjon til andre programytringer i deres historisk, sosialt og kulturelt skapte program- og maskinvare omgivelse. Gjennom å lage en logisk regel for objekter i programytringer, synliggjorde Nygård og Dahl sine programmeringsperspektiv både for seg selv og andre. De konkrete programytringene som realiserte SIMULA ble byggestener og bærere av dette synet, og deres stemme kan andre programmerere nå ta i bruk som en grammatisk basert stemme i sine ytringer.

Erfaringene fra vårt prosjekt bekrefter at denne påvirkningen fra programmereren på de grammatiske strukturene i programmeringsspråk kan være mer nærliggende enn vi er vant til å forvente. På samme måte som vi har kunnet utviklet nye ord og symboler, har vi også utviklet nye grammatiske strukturer. Vi fant f. eks. at vi trengte å

nyttiggjøre oss objekter og funksjoner run-time som vi design-time ikke hadde oversikt over. For å imøtekomme dette behovet etablerte vi først noen leksikalske enheter. Men da bruk av disse leksikalske ressursene så ut til å følge et fast mønster og ble brukt igjen og igjen, formelig tvang en ny syntaktisk, regel seg fram. Denne syntaktiske ressursen ble basert på de etablerte leksikalske enhetene og nye programytringer i en precompiler. Den leksikalske ressursen vi hadde etablert for å imøtekomme konkrete, situasjonelle behov, videreutviklet vi således til en syntaktisk ressurs som andre kan ta i bruk i dialog med våre programytringer (se. 108).

Vårt prosjekt viser dermed også hvordan leksikalske strukturer fungerer som midlertidige stillas i en gradvis grammatikaliseringssprosess. Gjennom en serie enkle, konkret programytringer basert på helt konkrete situasjonelle og sosiale programbehov, utviklet vi gradvis en syntaktisk stemme. Denne stemmen gir en mulig forståelse og løsning på et mulig problem, og andre programmerere kan i dialog med våre konkrete programytringer etablere en posisjon og relasjon som refrakterer denne stemmen i sine egne programytringer.

Grammatiske ressurser og programmereren påvirker hverandre gjensidig. Programmereren bruker grammatiske ressurser til å posisjonere sine programytringer i forhold andres, og ved å bruke konkrete, grammatiske ressursene etablerer programmerere både ytre og indre relasjoner mellom sine programytringer. Samtidig etablerer programmereren nye mønstre i sine programytringer, og over tid kan disse designes og implementeres som nye grammatiske regler. Grammatiske ressurser kan dermed utveksles, refraksjoneres og orkestreres som programmereres stemmer.

Heteroglossia i programmering

Innen en dialogisk tilnærming forstås programmering som konkret forankrete, sosiale prosesser der leksikalske, grammatiske og konvensjonelt bårde stemmer posisjoneres og relateres til hverandre. Det essensielle ved en programytring er derfor ikke en abstrakt, universell og homogen mening, men de konkrete, gjensidige og heterogene relasjonene disse ytringene etablerer til hverandre og hvordan disse relasjonene blir meningsfulle i en større sosial, situasjonell og kulturell omgivelse. Stemmene i programmeringsspråk medierer dermed programmerernes posisjon og perspektiv i relasjon til hverandre. Programmerere kan derfor heller ikke konstruere absolutte, perfekte og endelige programytringer, de kan bare forsøke å orkestrere, refraktere og harmonisere sine perspektiv med andres tidligere, samtidige og fremtidige stemmer, relasjoner og posisjoner.

Dette medfører at den systemiske virkeligheten som disse programytringene i både program- og maskinvare utgjør, også bør forstås som konkret, gjensidig og heterogen virkelighet. Programmene i denne virkeligheten er bygget opp som svar på historisk

skiftende situasjonelle, sosiale og systemtekniske krefter, og ettersom disse kreftene hele tiden er i utvikling i forhold til hverandre, er også den konkrete, sammensatte systemvirkeligheten i kontinuerlig utvikling. En nærliggende modell for å forstå denne virkeligheten og de situasjonelle, sosiale og kulturelle prosessene denne virkeligheten konstitueres av er derfor også heteroglossia (jf. s. 55). To iøyenfallende bilder på heteroglossia i programmering er utviklingen av det åpne operativsystemet Linux og det proprietære operativsystemet Windows.

Open-source prosjekter og da i særdeleshet utviklingen av Linux-plattformen er på mange måter heteroglossia satt i system (GNU 2003). "Linux"⁸ skapes av mange programmerere som fra hver sin kant ytrer program. Disse programmererne posisjonerer seg aktivt med og mot hverandre gjennom å direkte og indirekte relatere sine programytringer til hverandre. I denne massen av gjensidige posisjonerte program og programytringer kan programmerere og brukere avhengig av behov og preferanser selv bestemme hvilket utvalg av program og programytringer de ønsker å installere, bruke og programmerere i samarbeid med.

I tillegg forandrer og utvikler massen av programytringer seg hele tiden etter hvert som nye versjoner og program programmeres. Den foranderlige, amøbiske og grenseløse egenskapen ved "Linux" forsterkes også av at programmerere påvirkes av forskjellige situasjonelle, sosiale og systemtekniske krefter. Deres behov og preferanser kan derfor også stå i motfase med hverandre, og det er derfor heller ingen umulighet at motstridende symboler, logiske regler eller konvensjoner kan eksistere i konkurranse med hverandre innenfor denne løst avgrensede massen (se bl. a. Economist 2001a).

Beskrivelsen av Linux over synliggjør i særlig grad de sentrifugale, diversifiserende kreftene i "programheteroglossian" (jf. Bakhtin 1981: 270ff). Men også i Linux er det sterke sentripetale, forenende krefter (jf. *ibid.*) som sørger for en viss treghet i utviklingen og en stabilitet nødvendig for effektiv meningsskapning. Slike sentripetale krefter er det derimot lettere å få øye på i Windows, Linux rivaliserende operativsystem.

I Windows skjer det også hele tiden nyskapninger og eksisterer krefter som står i et heterogent forhold til hverandre, men i forhold til Linux er Windows og dets eiere Microsoft (MS) langt mer opptatt av faste grenser, universalitet og homogenitet. Dette

⁸ Linux omtales eller misforstås ofte som en større masse av programmer og programytringer enn det i "egentlig" er tale om (*ibid.*). "Egentlig" er Linux bare ett av flere program, én av mange stemmer. Men ettersom Linux er kjernen (programmet som styrer prosessoren) i operativsystemene som baserer seg på den, får den også en sentral, om enn ofte skjult og underliggende rolle som mange andre program på den samme plattformen indirekte må relatere seg til. Denne sentrale, svarende rollen medvirker, med rette eller ikke, til omdøping eller forveksling av hele programmassen med Linux.

skyldes hovedsakelig to brede økonomiske og sosiokulturelle egenskaper. For det første er bedriften MS en institusjon primært drevet av økonomiske interesser. MS' primære inntektskilde er salg av program, og dette kombinert med Windows posisjon som den primære plattformen for millioner av brukere og programmerere, gjør at MS kan tjene store penger på å få brukere av ett av deres program til å orientere seg mot andre MS programmer. Program og programytringene i Windows og andre MS program relaterer seg derfor ofte mot hverandre. Dette resulterer i både en universalitet og homogenitet disse programmene i mellom, samt en relativt tydelig grense mellom program som er inne i varmen og hvilke som ikke er det. Denne varmen av interne relasjoner er Windows.

"Microsoft can be seen as the dominant nation in the digital realm - just as America is in the material world. With Windows, its rulers control the largest, richest digital territory: a software platform on which others can build, with its say-so, applications out of bits and bytes. Like a superpower, Microsoft has good reason to defend, even extend, its sphere of influence: what the firm calls its 'ecosystem'" (Economist 2003).

For det andre orienterer MS seg mot brukere av program som i mindre grad enn Linux-brukere er opptatt av å posisjonere i den digitale virkeligheten. En større andel av disse brukerne er verken motivert eller forberedt på å ta i bruk nye program og versjoner som repositionere seg i forhold til andre program. Disse brukerne fører derfor et ytre press på MS om å opprettholde stabilitet og homogenitet i grensene i Windows over tid. Brukere av Linux regnes derimot av mange for å være mer proaktive i sitt forhold til nye program og oftere å programmerere selv. Kreftene i Linux trekker derfor sterkere i sentrifugale retninger.

Referanser:

- Agar M. & Hobbs J. 1982: Interpreting discourse. *Discourse processes* 5, 1-32.
- Bakhtin M. M. 1981: *The Dialogic Imagination*. Texas: The University of Texas Press.
- Bakhtin M. M. 1984: *Problems of Dostoevsky's Poetics*. Minneapolis, MS: The University of Minnesota Press.
- Bakhtin M. M. 1998: *Spørsmålet om talegenrane*. Bergen: Ariadne Forlag.
- Baetjer H. jr. 1998: *Software as Capital*. *IEEE Computer Society Press*, 85ff.
- Berge K. L. 1996: "Ulike perspektiver på taus kunnskap". Prøveforelesning for dr.art.-graden, NTNU. Upubl.manus.
- Clark K. & Holquist M. 1984: *Mikhail Bakhtin*. Cambridge, Mass.: Harvard University Press, USA.
- Chomsky N. 1966: Linguistic theory. Lester M. (ed.) 1972: *Readings in Applied Transformational Grammar*, 2.utg., 36-45. New York: Holt, Reinhart & Winston.
- Crowley T. 1990: That obscure object of desire: a science of language. Joseph J. E. & Taylor T. J. (eds.): *Ideologies of Language*, 27-50. London: Routledge.
- Dahl T., Sørensen K. H. 1997: "På langs og på tvers. Disiplin, profesjon og tverrfaglighet i det moderne forskningsuniversitetet". Dahl T., Sørensen K. H.(red.): *Perspektiver på tvers: Disiplin og tverrfaglighet på det moderne forskningsuniversitetet*, 9-17. Trondheim: Tapir Forlag.
- Darwin I. F. 2001: *Java Cookbook*. Sebastopol, CA: O'Reilly.
- Ecker B. 2001: *Thinking in Java (2nd ed.)*. www.mindview.net/Books/TIJ/, sist lest aug 2001.
- Economist 2001a: "Out in the open: The world is taking to open source". www.economist.com/displayStory.cfm?Story_id=568269, sist lest 30.8.2003.
- Economist 2001b: "Report: Programming: A lingua franca for the Internet". http://www.economist.com/displaystory.cfm?story_id=779459, sist lest 1.9.2003.
- Economist 2003: "Microsoft as America: As Colin Powell is to George Bush, so Craig Mundie is to Bill Gates". www.economist.com/displaystory.cfm?story_id=1940385, sist lest 30.8.2003.
- Encyclopædia Britannica Online 2003: "computer programming language". <http://search.eb.com/eb/article?eu=25460&hook=134611#134611.hook>, sist lest 30.8.03.
- Evensen L. S. 1986: Kapittel 1. Anvendt Språkvitenskap. *Den vet best hvor sko(l)en trykker....* Avhandling for graden doctor artium, Universitetet i Trondheim.
- Evensen L. S. 2002a: Fornyelse i Tåkeheimen: Kommunikasjon på tvers av faggrenser. Skretting K., Olstad L.(red.): *Forskning på tvers: Tverrfaglige forskningsprosjekter ved NTNU*, 133-144. Trondheim: Tapir Akademisk Forlag.

- Evensen L. S. 2002b: Convention from below: Negotiating interaction and culture in argumentative writing. *Written Communication* 19(3), 382-413.
- EiT 2003: "Ekspert i Team: tverrfaglig prosjekt". NTNU, <http://www.eit.ntnu.no/>, sist lest 15.04.03.
- Faarlund J. T. 2002: Grammatikalisering av ordstilling. Moen I. et.al.(red.): *MONS 9: Utvalgte artikler fra Det niende møtet om norsk språk i Oslo 2001*. Oslo: Universitetet i Oslo.
- GNU 2003: "the GNU Project". www.gnu.org, sist lest 30.8.2003.
- Hopper P. J. & Traugott E. C. 1993: *Grammaticalization*. Cambridge: Cambridge University Press.
- Greenberg M. J. 1974: *Euclidean and Non-Euclidean Geometries: Development and History*. San Francisco: W. H. Freeman and Company.
- Holmevik J. R. 2003: "The History of Simula". <http://java.sun.com/people/jag/SimulaHistory.html>, sist lest 30.8.2003.
- Holquist M. 1990: *Dialogism (2nd ed.)*. London: Routledge.
- Haavind H. 2000 (red.): *Kjønn og fortolkende metode: Metodiske muligheter i kvalitativ forskning*. Oslo: Gyldendal Akademisk.
- Hoel T. L. 2000: *Skrive og Samtale: Responsgrupper som læringsfellesskap*. Oslo: Gyldendal akademisk.
- Jacobson I., Booch G., Rumbaugh J. 1998: *The Unified software development process*. Reading, Mass.: Addison-Wesley.
- Javadoc 2003: "Java 2 Platform, Standard Edition, v 1.4.1 API Specification". <http://java.sun.com/j2se/1.4.1/docs/api/>, sist lest 25.9.2003.
- Jennings N. R. 2000: On agent-based software engineering. *Artificial Intelligence* (117), 277-96.
- JCP 2003: "Java Community Processes". www.jcp.org, sist lest 30.8.2003.
- Kline M. 1972: *Mathematical Thought from Ancient to Modern Times*. New York: Oxford University Press.
- Lakoff G., Johnson M. 1980: *Metaphors we live by*. Chicago: University Press of Chicago.
- Lewis J., Loftus W. 1998: *Java Software Solutions: Foundations of Program Design*. Reading, Mass.: Addison-Wesley.
- Mathisen W. C. 1997: Erfaringer med tverrfaglighet fra forskningsmiljøer utenfor og i periferien til universitetssystemet. Dahl T., Sørensen K. H.(red.): *Perspektiver på tvers: Disiplin og tverrfaglighet på det moderne forskningsuniversitetet*, 29-37. Trondheim: Tapir Forlag.
- Miller C. R. 1984: Genre as Social Action. Freedman A. (ed.) 1994: *Genre and the new rhetoric*. London: Taylor & Francis.

- NESH 2001: "Forskningsetiske retningslinjer for samfunnsvitenskap, jus og humaniora, vedtatt 15.2.99". Den nasjonale forskningsetiske komité for samfunnsvitenskap og humaniora, <http://www.etikkom.no/NESH/forskretn.htm>, sist lest 12.11.01.
- Nystrand M. 1986: *The Structure of Written Communication*. New York: Academic Press Inc.
- Nystrand M. 1989: A Social-Interactive Model of Writing. *Written Communication* 6(1), 66-85.
- Nystrand M., Wiemelt J. 1991: When Is a Text Explicit? Formalist and Dialogical Conceptions. *Text* 11(1), 25-41.
- O'Hare G. M. P. & Jennings N. R. 1996: *Foundations of Distributed Artificial Intelligence*. New York: John Wiley & Sons.
- Pennycook A. 2002: Language and linguistics/Discourse and disciplinarity. Barron C., Bruce N. & Nunan D. (eds.): *Knowledge and Discourse*, 13-27. London: Longman.
- Polanyi M. 1962: *Personal Knowledge: Towards a Post-Critical Philosophy*, corrected edition. Chicago: The University of Chicago Press.
- Pressman R. S. 2000: *Software Engineering: A Practitioner's Approach: European Adaption (5th ed.)*. London: McGraw-Hill Publishing Company.
- Reddy M. 1979: The Conduit Metaphor. Ortony A. (ed.): *Metaphor and Thought*. Cambridge: the University Press.
- Rommetveit R. 1974: *On message structure: A framework for the study of language and communication*. New York: John Wiley & Sons.
- Rommetveit R. 1992: Outlines of a dialogically based socio-cognitive approach to human communication and cognition. Wold A.H. (ed.): *The dialogical alternative: Towards a theory of language and mind*, 19-44. Oslo: Scandinavian University Press.
- Saussure F. 1916: *Cours de Linguistique Générale*, Payout, France. Eng. Utg.: The Philosophical Library, 1959; McGraw-Hill, 1966.
- Sørensen K. H. 1997: Den nye tverrfagligheten. Fra polytekniske generalist til polyvalent spesialist. Dahl T. & Sørensen K. H.(red.): *Perspektiver på tvers: Disiplin og tverrfaglighet på det moderne forskningsuniversitetet*", 19-28. Trondheim: Tapir Forlag.
- Sørensen K. H. 2002: Tid for tverrfaglighet?: Jakten på vennligsinnede spesialister. Skretting K. & Olstad L.(red.): *Forskning på tvers: Tverrfaglige forskningsprosjekter ved NTNU*, 7-18. Trondheim: Tapir Akademisk Forlag.
- Schwebs T. 1994: En ny skrift. Schwebs (red.): *Skjermtekster: Skriftkulturen og den elektroniske informasjonsteknologien*, 1-14. Oslo: Universitetsforlaget.
- Strømhylden A. 2002: *Interactionistic agents: ...a fresh perspective on multi agent systems*. Upubl. Diplomoppgave ved IDI, NTNU, Trondheim.
- UiO 2003: "Forskning med tellekanter". http://www.admin.uio.no/oepa/budsjett/finansieringsmodell/publiseringsutv_innstilling.rtf, sist lest 1.10.2003.

- Voegele J. 2001: "Programming Language Comparison".
www.jvoegele.com/software/langcomp.html, sist lest 1.9.2003.
- Vygotsky L. S. 1978: *Mind in Society: The Development of Higher Psychological Processes*, 19-75. Cambridge, Mass.: Harvard University Press.
- Vygotsky L. S. 1997: *Thought and language*, (ed. A. Kozulin). Cambridge: MIT.
- Wertsch J. V. 1991: *Voices of the Mind*. Cambridge, Mass.: Harvard University Press.
- XP 2003: "Extreme Programming: A gentle introduction".
<http://www.extremeprogramming.org/rules/pair.html>, sist lest 25.5.03.
- Ørstavik I. T. B. 2001: "IKT-systemutvikling som en dialog". Upubl. hjemmeeksamen i HFAVS305: Metoder i anvendt språkvitenskap, NTNU, Trondheim.
- Ørstavik I. T. B. 2002a: "SjangDAI: Problem oriented tool". Upubl. mappeinnlevering i HFAVS306, NTNU, Trondheim.
- Ørstavik I. T. B. 2002b: "SjangDAI posters: Internal use of language in agents". Upubl. design-poster tilknyttet SjangDAI-prosjektet, NTNU, Trondheim.
- Ørstavik I. T. B. 2003: "SIMAS: Social interactionism as an approach to multi agent systems". Upubl. søknad til NFR fra ISK, NTNU, Trondheim.