An Interview with

TERRY ALLEN WINOGRAD

OH 237

Conducted by Arthur L. Norberg

on

11 December 1991

Stanford, CA

Charles Babbage Institute
Center for the History of Information Processing
University of Minnesota, Minneapolis

Terry Allen Winograd Interview
11 December 1991

Abstract

Winograd describes his education in computer science and introduction to linguistics at the Massachusetts Institute of Technology (MIT). He discusses the work of Marvin Minsky and other in artificial intelligence. He describes his move to the Stanford Artificial Intelligence Laboratory and his additional linguistic research at Xerox-PARC. Winograd compares the approach to artificial intelligence at MIT and Stanford. He describes his involvement with obtaining funding from the Information Processing Techniques Office of the Defense Advanced Research Projects Agency.

TERRY ALLEN WINOGRAD INTERVIEW

DATE: 11 December 1991 INTERVIEWER: Arthur L. Norberg

LOCATION: Stanford, CA

NORBERG: I'd like to ask you to say a couple of things about yourself. I know from *Who's Who* entries that you went to school in Colorado first, and then to MIT. What is your background before you went to M.I.T.?

WINOGRAD: I grew up in Colorado, and went to a private liberal arts college there called Colorado College, which considers itself one of the sort of Midwestern good liberal arts colleges, like Grinnell, Knox, Carleton - that kind of school. And I was formerly a math major, but in a liberal arts college of that kind that meant I took one math course every quarter. Other than that I did all sorts of things, took courses in all subjects, and was editor of the college newspaper and just the general things you do as a liberal arts student.

NORBERG: When you arrived at MIT and began on your dissertation work, who was there at the time? Who were the graduate students, what faculty was there?

WINOGRAD: Well, the dominant faculty - and of course, MIT - you know the layout there, right? There was a big division between the fifth floor and the eighth floor, as it was thought of -

NORBERG: Fifth floor being AI -

WINOGRAD: No, eighth floor was AI -

NORBERG: Just the opposite.

WINOGRAD: - and fifth floor was what is now LCS, but at that time had a different name. So people like Fano and Corbato and all those people were fifth floor. And we had almost *no* contact with them. So the people I had contact

with were the AI people, of which Minsky and Papert were the center of everything. Mike Fischer was a young faculty member. I honestly don't remember other faculty in that area - it'll probably come to me. The first wave of AI graduate students had already left; people like Bobrow and Raphael - Joel Moses was on the faculty. I took my first LISP course from Joel Moses. Martin - what was his first name? He died quite young. But Moses and Martin were the two young faculty in the department - Bill Martin.

NORBERG: Bill Martin, yes.

WINOGRAD: And so the group of graduate students who were around when I was included Winston, Sussman, Charniak, Waltz - a guy named Tom Jones was doing an infant simulator. And then, of course, a lot of the structure was not the graduate students but the hackers. So on the *ninth* floor, which is where the machines were, there were people like Greenblatt and Knight and Holloway and Rolly Silver, and people like that.

NORBERG: Okay. What sort of problems were under investigation, as you remember them now?

WINOGRAD: When I got there, there happened to be a burst of interest - it was, I think just after Minsky and Papert had written their perceptrons book, so there was a burst of interest in cellular automata. The first thing I did that actually got me some status there was to - I went to a lecture of Papert's in which he described a conjecture about something with regard to cellular automata, and I came back with a proof of it. So, I know there was a student there by the name of Terry Beyer, or something like that, who was doing stuff. So, one of the areas - but it was a sort of a side area - was this notion of what you could do with cellular automata. Then there was the robot project, which was the big funded DARPA project - ARPA project, in those days. I don't remember...

NORBERG: Can we go into that a little bit more deeply, though? When you say "the robot project," what did that mean? Was it split up into parts, or was everybody working on the same thing?

WINOGRAD: It was a collection of fragments. There were different arms. For example, I remember there was a thing

that was called the Minsky arm, which was a multi-jointed lobster claw kind of arm, which was being experimented with and had a prosthetic hand that actually looked like a hand on the end of it, sitting on - you know, hanging from the ceiling in one place. Over next to it, then, was one of the more traditional, you know, three degrees of freedom, standard industrial kind of arm. There was not - I remember - I'm trying to think - the notion of getting a robot to play ping pong, for example, was not - I think it had already been sort of - they weren't focusing on it. I don't remember that being active; I remember it being talked about, but never being active. It was basic hand-eye coordination. What I remember is their focus was hand-eye; could you figure out the coordinates it takes to get the thing over there and pick something up reliably and get the position right, and so on, mostly using wood block kind of structures to do it. But not as a focused project. There was just lots of people hacking various arms, and various interfaces for the arms, and various schemes for calculating position, and so on. It was all hand-eye; I don't remember any mobile - you know, robot moving around. It was all hand-eye. And there was a lot of vision work. Guzman was still there when I got there, and then Waltz worked on vision and Tom Binford, who's here now, was there at the time. A variety of people working on line finding and that kind of stuff, low level machines, plus modeling the joints and the higher-level modeling based on [?]. So, certainly robot meant hand-eye, and there were the two pieces: The vision work and the manipulator work, basically.

Then there were - I would guess the best way to describe it is "pet projects." For example, I did work on language, but it wasn't a project. I just came in and said, "I'm interested." I had done a year of linguistics on a Fulbright before I went to MIT, and I wanted to work on that and, basically, the attitude was if you had a smart student who had an idea, let him work on it, right? There was no sort of top-down structure.

NORBERG: Where did you do the Fulbright?

WINOGRAD: University College in London. It was a sort of a boondoggle. Basically, I was a senior in this liberal arts college, and people said, "You should go overseas. It's a great," you know, "it's a great way to get experience." And I said, "Well, I'm interested in computers." And, in my ignorance in those days, I said nobody outside of the United States has anything in computers. I didn't realize that they had *invented* them in England, right? That's

another question. The only language I had taken in college was Russian. And in those days you didn't go to Russia on a Fulbright. So, I needed to go to an English-speaking country to learn something related to computing but that wasn't computing. And I literally thumbed through catalogs and found this sort of interesting linguistics group in London, which turned out to be very relevant. I mean, it turned out to be a major piece of what I did later but, in fact, the planning was really "here would be a good experience for a year."

NORBERG: I'm sorry, I didn't mean to deflect you but I wanted to make sure whether it was relevant or not.

WINOGRAD: And Minsky was very interested in language, so he encouraged me. he encouraged Gene Charniak and his children's story work. Dave Waltz was actually working on language at that point. He did the vision stuff later. He started off trying to do a language project. But it was individual efforts; it wasn't a coherent - you know, it wasn't a *project* in any larger sense. And then there was the planning. Carl Hewitt was there. That's the other person who was very relevant to my work because, when I started trying to do the language, it became clear you needed some kind of a planning/problem-solving platform on which to do the question answering, and he had developed his ideas for Planner. Planner was one of the systems that was always about to be done implementing. [laughs] There were *years* during which Planner was going to be the wonderful thing to use, but didn't *quite* work yet, and so on. And, I decided it was exactly the piece I needed for that stuff, and that's when Sussman and Charniak and I got together and did MicroPlanner, which was intended as a, you know, quick and dirty get something that does more-or-less the right thing until the real thing exists.

NORBERG: Was there some pressure to do that, to provide some sort of application, in a sense? To implement a program, or implement a language?

WINOGRAD: Well, certainly implementation was the coin of the realm. There was nobody doing a thesis which was "here's a bunch of ideas, period." It was "here's a program that runs - and I have some ideas to lend to it."

NORBERG: Well, I was just thinking of Shank's recent piece in *AI Magazine* -

WINOGRAD: I haven't read it.

NORBERG: - in the winter, where he says that one of the problems associated with AI is that, in the past, anyway, until inference engines were developed, that most theses were simply investigations of good ideas, but they never got to the implementation stage, because that wasn't the point of the dissertation in the first place. And so, we never learned, "we" being the AI people, we never learned how to develop a product in the normal sense of what a product would mean on the market.

WINOGRAD: Well, implementation and product are two stages. That is, implementation was always there as the coin of the realm. Implementation meant something you could show off. It didn't mean something that somebody else could use. So - I mean, my thesis was certainly that way. I mean, you know, the famous dialogue with SHRDLU where you could pick up a block, and so on, I very carefully worked through, line by line. If you sat down in front of it, and asked it a question that wasn't in the dialogue, there was some probability it would answer it. I mean, if it was reasonably close to one of the questions that *was* there in form and in content, it would probably get it. But there was no attempt to get it to the point where you could actually hand it to somebody and they could use it to move blocks around. And there was no pressure for that whatsoever. Pressure was for something you could demo. Take a recent example, Negroponte's Media Lab, where instead of "perish or publish" it's "demo or die." I think that's a problem. I think AI suffered from that a lot, because it led to "Potemkin villages", things which - for the things they actually did in the demo looked good, but when you looked behind that there wasn't enough structure to make it really work more generally.

NORBERG: Is that a question of size, or is it a question of the idea itself - the idea's too small?

WINOGRAD: The idea of - ?

NORBERG: Well, let's say the ideas behind the blocks world, SHRDLU.

WINOGRAD: Well, I think it was based on a presupposition - at least an attitude - that making things work in the large, really working, was just like getting a demo except more details to be filled in. That is, if you had a basic idea and you could show it worked on something then it was just a sort of grubby, detail work to fill in all, you know, the hundreds of entries you would need to make it work for real. But that - an idea - and this is tied to the top-down, rationalistic way of approaching it, right. An idea which said, "here's a nice logical way this should work - would work in practice if you just went far enough with the details." And I think that's been a problem with AI all along. It's true in problem-solving, right? Problem-solving, as conceived by Newell and Simon and developed, and so on, has a certain realm of applicability but it's very different from, you know, you coming to me and saying, "I have a problem. Would you help me solve it?", in terms of - to take the most obvious things - the hard part is figuring out what the problem space *is*, not searching it.

NORBERG: I'll come back to that when we get to the general questions. Back to the work in the AI lab . . . you were talking about your work with Charniak and Guzman and so on. What else was going on besides those two projects? You mentioned the -

WINOGRAD: There was the robots -

NORBERG: - hand-eye, and you now mentioned the language work that you became interested in.

WINOGRAD: Right. Pat Winston was doing his thing on learning, which was pseudo-robotic. I mean, he used blocks in his model, but it wasn't really tied to the robot. There was a lot of technical development, of course. Text editors and time-sharing systems and all that stuff, which was done not by the graduate students, although everybody ended up getting involved to some extent.

NORBERG: Was this *in* the AI lab now?

WINOGRAD: Yes. The AI lab - remember, there were two time-sharing systems being built. That is, everybody - what was there when I got there was ITS - no, I'm sorry -

NORBERG: No, CTSS.

WINOGRAD: CTSS, which was what I used for course work and so on when I first started learning LISP. On the fifth floor Multics was being developed. On the ninth floor, which belonged to the AI lab, was ITS, and there was, you know, it was like the Hatfields and the McCoys, right. You were either a winner or a loser, depending upon which side you were on. There was no question about that. And, actually the fifth floor people owned part of the ninth floor, too. There was this wall half-way down the middle. There was the AI...

NORBERG: Literally?

WINOGRAD: Yes, there was a door, actually. There were actually these two separate rooms. All the Multics stuff was at one end of the floor, and all the AI stuff was at the other end of the floor, and basically it was like two separate floors; it wasn't a connected space. And what was going on there was the development of ITS. I assume you know the history of ITS?

NORBERG: Yes.

WINOGRAD: I'd be glad to fill in details on any of this stuff that you *don't* know.

NORBERG: Well, please do. I'm going to play ignorant, because I want you to tell me what you know. I don't want to be telling you what I know.

WINOGRAD: Right. Well, ITS was the descendant of the PDP DEC systems - time-sharing system - the PDP-1 and actually, probably the TX - whatever it was before that, I don't know.

That was all before I got there. By the time I got there, they had a PDP-6, and they were programming - and had developed - they were using - it was already far enough along to use - the ITS, which was being done by Greenblatt and Knight and Nelson and that group, with certain strong ideological assumptions which were very different from the ideological assumptions of Multics. For example, Multics had 32 protection rings and ITS had *no* protection of anything. Anybody could read and change any file in the system, and that was not just an accident. That was a position. Multics had all sorts of, you know, reliability things, and ITS anybody could get into the mode where they could simply stop the system. And so it was a very strong socio-cultural difference, where the Multics had this sort of notion of a utility for the world, and ITS had the notion of a playground for a bunch of smart hackers who were a community, and therefore...

NORBERG: Now, were there any implications of that, however, when you think about someone coming in the following morning after a group has been working all night, and they've changed a number of files in the system, and you've been trying to develop something and get damn mad about having your file changed. Did any of those things ever occur to your recollection?

WINOGRAD: Not really. It was a small enough community and the entry fee was being pretty sophisticated. I mean there was nobody using it who wasn't sophisticated. And there was a code of honor in some sense. Nobody messed around with programs. If I was writing a program, nobody would ever change my files. There was a problem with no protection. There was no protection, but nobody was doing that.

NORBERG: All right.

WINOGRAD: And, of course, it was also in a day when communications were different, so nobody was dialing in from outside. You had to actually *be* on the ninth floor, sitting among the terminals. When you did call a command which brought the system to a halt, somebody yelled, "Hey! What happened to the system?" We were all sitting in one room. And I think that model works for a community of that size in one geographic location, because then you

do have all of the human, natural social interactions and controls which operate. And I think it was a very sad and tough lesson for the people who were really devoted to that, including Stallman and that group, as they began to see their wonderful system opened up to a larger world, and that just didn't work anymore. I mean, any high school kid around the country could dial in to the AI system on INTERNET, and you could no longer trust the fact that people wouldn't mess around with your files.

NORBERG: Now what were the implications of that, other than this small group, community, sort of splintering, perhaps?

WINOGRAD: Well, that didn't happen until later. The NET really didn't become widely available until after I left MIT. It was there, but, you know, you could talk to your friends at the other AI labs. That was about as far as it had gone by the time I left there in '73.

NORBERG: Okay. Now, you started that by mentioning that time-sharing was one of those areas that was going on at the time.

WINOGRAD: If you said, "What was the conversation around the hallways?", it was advances in systems. I know that - what's his name? - Peter Sampson was developing TECO, which was the -

NORBERG: That was T-E-C-O, wasn't it?

WINOGRAD: Yes. Text Editing and Corrector, or something, which was the cleaned-up version of his original program called the string-handling interpretive translator, which is an ancestor, of course, of a lot of word processing stuff. But again, it had a very particular flavor of [?]. That's what I used when I wrote my dissertation. In fact, to print my dissertation I had to get somebody up there to install a new hardware device. I was the first person, I guess, who really wanted to write something using the word processing system and print it on something nicer than a line printer. Of course, if you have a line printer, that's fine, but how do you print on something nicer. So, I actually

got them to install an IBM - one of those - I forgot what it was called. It uses a Selectric mechanism but had a terminal.

NORBERG: Now, what was the interaction among this group? It sounds to me like, as I've heard it from others, that there were a group of very sophisticated graduate students who were working on various problems, who may or may not have seen people like Minsky or Papert during the course of this work.

WINOGRAD: Right, you'd see them once in a while.

NORBERG: They were sort of all - I don't want to say "working together" - I don't want to go that far. But, they were sort of developing something on their own. It wasn't a case of dealing with a faculty member's research area or domain, and sort of working on a piece of that. Is that a correct interpretation, from your recollection?

WINOGRAD: It was people who were basically working on their own and then knowing enough about what each other was doing to join up. So, for example, this stuff on Micro-Planner was useful for all three of the people who did it, so we got together and did that. And to comment, and so on. I probably saw Marvin - well, he wasn't actually my official advisor, Seymour was, but I don't think I saw either of them to talk about my dissertation more than once every few months. It wasn't like they were directing the project.

NORBERG: How about course work, or ways of learning. What were the prevailing notions of methods and approaches to AI?

WINOGRAD: There was none of that. Course work was something you had to do to satisfy the MIT academic system. So, people took the courses they needed to to get them through that. But I think it was treated basically as, not that you were going to learn what was relevant to your research in courses; you took courses to get your degree. It was very much of an attitude - and I think Minsky still has this to a large extent - of nothing that anybody did before is really of much interest. We've invented everything here.

NORBERG: I won't ask you to comment on that. I have my own view of that, however. How did you become interested in that problem of natural language and developing what I will call for short here, "the blocks world?" I realize it has a predecessor history, and we'll get to that I assume.

WINOGRAD: Well. I can tell you various strands, and of course you never know what motivations actually lead to things. I *do* remember one of the first things I ever read about AI was an article by Minsky in *Scientific American* "Steps Toward Artificial Intelligence", which talked about Bobrow's program. And I remember being impressed by that, so I think the notion that you can get computers to use language was intriguing.

NORBERG: Would that be "Steps"? "Steps" first came out in '61, and then it was reprinted several times, of course. It was reprinted again in the '63 book by Feigenbaum and Feldman. I don't remember it being in *Scientific American*, though.

WINOGRAD: Maybe not. I don't remember.

NORBERG: There was something in *Scientific American* as a result of the conference that was held in about '66, but I don't think it was "Steps".

WINOGRAD: Maybe it wasn't. Maybe it was a different one.

NORBERG: So, I'm trying to make a connection to where Bobrow's work would have been discussed.

WINOGRAD: Well, it would have to be that late, because Bobrow didn't do his work until '64.

NORBERG: Right. How about *Semantic Information Processing?*

WINOGRAD: No. I mean, I know the book, but I didn't see it before I got to MIT. It was a popular magazine article.

It may not even have been in *Scientific American.* It was some popular magazine article about AI. And I'd always just been interested in language. I just happened to like it. And, as I say, I had this opportunity to study abroad for a year so I went into linguistics. If you look at the application I sent to Fulbright it said, "Language is important and computers can do it, and maybe we should try to connect them." I mean, I was certainly looking towards that connection without really having much of a sense of what it was.

NORBERG: How did the problem idea develop when you got to MIT?

WINOGRAD: What I remember, vaguely, is first of all, I'm sure what happened is I went in and talked to Minsky or Papert - whoever it was I talked to first about this - who said, "Oh, well, we're interested in having computers do language". They had, you know, *Semantic Information Processing* had been done, and there was Raphael and Bobrow and so on - "we want to go farther in that." And Minsky's idea at that time, which he was pushing very hard, was that the right thing to do was children's stories, because they were simpler. And Charniak ended up going in that direction. In fact, I think Waltz was also using children's stories when he was doing natural language. It didn't seem right to me.

NORBERG: Why not?

WINOGRAD: I don't know. It just seemed like it depended too much on knowing the things that children know. The knowledge was too deep. It's hard for me, right now, having 20 years of perspective to say, "I was seeing the following points. But I just know that I very strongly resisted. I actually went through - I remember taking a children's encyclopedia - not children's stories, but an encyclopedia written for children, and trying to think about what it would take to write a program that would understand that, and not feeling comfortable with that. I'm sure as I say, in hindsight I can see deep philosophical underpinnings, but at the time it didn't strike me that children's stories were the right thing. And, of course, the blocks world was in the air there. The hand-eye coordination people were using it, so the people who were doing vision were looking at scenes full of blocks. The people who were doing hands were having to pick up blocks, and so on. So, at some point, and I honestly don't remember the history of this.

Somewhere I have old file cards and stuff that I made notes on, but some...Oh, I know what else I thought about. Bill Woods had done a system on airline reservations. I remember thinking through examples of something like an airline reservation system. And at some point it just seemed like it would be a natural thing to use the blocks world. Not to use the actual software, but just as an example that people were familiar with, probably with the idea that it would be nice to connect it to the computer because it would be a great demo, if you had it running. I think I definitely felt that the visual - the fact that you could talk about something you could actually see was an important thing for getting people to pay attention to it.

NORBERG: Rather than children's stories.

WINOGRAD: Rather than children's stories. You could actually see the blocks move, and so on. It was at one level more concrete than something where you would just ask a question and get an answer. You actually could talk about something that was happening. So I decided to do it.

NORBERG: Now, what did the problem require once you had made that decision? What sorts of things did you set out to do? What I'm looking for here is to get behind the dissertation and behind the subsequent publication on natural language understanding and so on, to see whether we can reconstruct any of your both approaches - those that worked and those that didn't, let's say, as a theoretical construct - and also some of the motivation that went into it, *and* how you got some help from the other graduate students with those motivations.

WINOGRAD: Well, going into it, I know that I was fairly fascinated with syntax. And of course, I had just spent a year in a linguistics department, which was basically syntax. And of course at MIT, there was a huge amount of syntax, but that was another one of these sort of feuding worlds; you couldn't talk to Chomsky and Minsky both, it was impossible. The one contact I had is I went and took a course from Chomsky, wrote a term paper saying why I thought this AI-ish approach would work, and he gave me a 'C' on the term paper. That was the end of my contact with the Linguistics Department.

TAPE 1/SIDE 2

NORBERG: We were talking about your -

WINOGRAD: The problems in syntax.

NORBERG: That's right.

WINOGRAD: One of the things I was really focusing on was how to apply some of the symbol-processing ideas to syntactic analysis, and that led to this thing I called *systemic grammar*, a form of grammar that I got from Halliday. Then I modified it into what I called procedural grammar or something, and tried to make it operational so it actually was a grammar and a parser sort of rolled into one. It was before the days when there was much sophisticated parsing theory, so it was fairly *ad hoc*, but effective. It was complex *ad hoc*. I remember being very interested in the question of syntactic-semantic mapping. I remember filling notebook pages with examples of quantifiers building.

NORBERG: What does that mean?

WINOGRAD: "Find a red block" - what's a good example? - "Put three red blocks on two boxes." Does that mean "put three on each of two boxes," or "put three altogether on two," and so on? And just going through lots of examples and realizing the degree of ambiguity that was resolvable by context. But in general, there were examples like, you know, "there were six vehicles with four wheels"; you know that you mean "six four-wheeled vehicles," as opposed to "six vehicles sharing four wheels." But if you say, "There were six people with four pizzas," right, you mean "there are six people all sharing four pizzas." And that has to do with knowing something about vehicles and wheels and pizzas, rather than syntax. So, the question that I was really focusing on is how do you make use of the world knowledge, so-called, to dis-ambiguate constructs in natural language which could have more than one interpretation. This quantifier one was an example which was fairly clean in a sense that you had a small, limited number of possible interpretations. It wasn't open-ended, right? You knew it was either four divided by six or four

for each of six, and you had to pick which of those two it was. The same kind of problem came up with pronouns. When you see the word "it," what does it refer to? How do you know? Well, to do that I needed to - first of all, I worked with a bunch of examples: "Pick up the block and put it on the table." Well, "it" means the block. But is the block bigger than one that you picked up before? Is *"it"*? Now, does it mean the one that you picked up before, or the one that is bigger? And the style, the methodology, was basically to come up with a particular area like that, to just sit down and write out lots of examples to try to get a feel for why, in one case, you had one answer and in a different case you had a different [answer], and then to come up with mechanisms which were fairly impromptu. I mean, it wasn't like somebody doing systematic theorem proving, or something. Okay. But clearly, recency is important. If something is more recent, you're more likely to mean that. So, what if we keep around a list of how recently things were mentioned? So, something which is farther up on that list, gets a little extra point. But, it's also not always that, because if it's the subject of the sentence, it's more likely to be it than the object, so we need to keep around the syntactic structure. So, the driving problem was this question of how do you use extra information, part of which is textual - what's mentioned recently - and part of which is world knowledge - like pizzas and vehicles, and so on - to dis-ambiguate natural language utterances into clearly-defined, you know, I mean *that* block and put it *that* place. I wasn't particularly using any background of linguistic theory or philosophy of language theory. I learned all that stuff later, right after I left. And that was the spirit, very much; you saw a problem, you came up with a few examples that gave you ideas, and you programmed a mechanism that handled those and then you started debugging. Minsky said - you know, one of his famous quotes is - "a program is just an empty page that needs to be debugged."

NORBERG: Well, there were a couple of things in that that puzzled me. One of them is that thinking back to the book, *The Hackers*, which was done 10 years ago now, in which one of the people, I recall, was quoted as having said in looking at such questions as how do you dis-ambiguate certain kinds of statements in order to let the machine be able, then, to make some conclusions about the next sentence said that the only thing that he had to rely on was what was going on in his own head. When I think about parsing structures and the way in which one tries to understand the construction of a paragraph that one is reading, we have quite a number of rules that we have learned over the years through various educational schemes. How did you bring all that to bear on this problem? I assume

you knew these things and maybe more of them than anyone else.

WINOGRAD: Well, the problems I was focusing on were the ones that you don't learn. If I say to you, when I use the word "it" in a sentence, how do you know which of the things previous I'm talking about, you can't give me a rule you learned in grammar school.

NORBERG: No, I can't. That's true.

WINOGRAD: You didn't learn a rule about that. It's just that whenever people use the word "it" in certain ways, you learn to do it right. Do *it* right. [laughs] So, it was clear to me. Of course, this also was true in linguistic theory. If I say, "How do you know what the structure of a sentence is," it's clear that what you learned when you learned to diagram sentences in school and so on, happened long after you knew how to understand English.

NORBERG: Yes.

WINOGRAD: So, the learning of those rules in schools wasn't what enabled you to understand English. It may have gotten you some additional stuff, but it's clear that a five-year-old understands quite complex sentences.

NORBERG: But what I'm driving at here is, you and I, in whatever similar ways, know how to do that because we learned it through listening to other people and testing the language ourselves, and so on. But, to go from that to both a syntactic structure and a semantic structure that will work in a program, it seems to me, is a rather major step. How does it happen? What is happening to make that step?

WINOGRAD: See, there are two answers. In hindsight, I will say, "I agree, that's a major step." I think the sense that was dominant then, and I was operating under, certainly, is that there are all sorts of things which are not available directly to conscious introspection, but, if you *were* able to examine them, would be more-or-less straightforward applications of symbol processing. So, although I don't know what goes on in my head when I hear a sentence - I

can't think hard and think about how I [?] sentence. If, in fact, I could do that what I would see, so goes the story, right, is something like, O.K., first I find the noun, then I find the verb, then I see if they match in features, and if they don't match in features then I do this. And you take the kind of thing a clever programmer would do given an algorithmic task like that, and the assumption is that's what's going on. So the problem is only to devise the right one, not to see a deep problem involved. You know, you may have trouble finding it. It may take awhile. It may be complicated. Also I think there was a certain aesthetic involved - I mean, it's driven by physics. It says, Okay, there's messy stuff, but there's also nice, clean, simple stuff happening underneath. So, even though real objects in the world move in all sorts of complicated ways, billiard balls on frictionless surfaces don't do that. And that, therefore, if you come up with the algorithm for the billiard ball on the frictionless surface, you'll later be able to patch on the things that handle all that other stuff.

NORBERG: That's what physicists say, that's right. And often they can't do it.

WINOGRAD: Well, for a certain class of problems they've done it very well. Of course that's what's seductive about it.

NORBERG: Yes. I was trained as a physicist; that's why I make that remark. Now while you were *doing* this then, were you conscious of the *need* to develop - I'm not saying this right, but - while you were investigating the problem and trying to develop the SHRDLU program, were you conscious of trying to find such an algorithm in such a way that you can now say "the steps were the following," and then repeat what those steps are for me? Or was it -

WINOGRAD: Yes. Now, I'm not clear of the thrust of the question. Certainly, you see the test of the idea was the program. The test was that I could build something which actually used that algorithm and -

NORBERG: What is that algorithm?

WINOGRAD: It was a bunch of stuff which put things on to recency lists and built syntactic structures and stored

syntactic structures and called programs that did planning and went back and tied them. It was no - it wasn't an essence, it was a bunch of mechanisms.

NORBERG: All right. And the test of the program, then, is whether or not, as I'm looking at the screen, the actual events that happen as the image moves blocks and so on is the test of whether you succeeded in developing that algorithm?

WINOGRAD: Right.

NORBERG: What sort of assistance did you get from others in the lab? And I don't mean assistance here in the sense of you asking a question and getting a direct answer, but what interplay was going on between you and the people who were trying to do this with actual mechanical arms?

WINOGRAD: None.

NORBERG: None?

WINOGRAD: The interplay was - two things. One, as I said, it was clear that some sort of generic inference engine was needed. You needed to be able to say "If the block is red, it's not green." Right? And you don't want to do that *ad hoc* for every possible question, so you needed something that today I would say is basically a simple application of the Prolog system. The basic things you get out of just a few predicates and connectives. And that was something that had enough generic use that it was clear it wasn't just what *I* needed and, as I said, that's where I got together within this project Micro Planner with Charniak and Sussman. So that we could all use it for our dissertations. And other people ended up using it for other things after that. So that was really a joint work. On the linguistic side, other than that, I was really pretty much working on my own. There was very little discussion about that. On the technical side, there was always a tremendous amount of back-and-forth with the people who were developing the LISP system, for example, were developing the display. I wanted to do this stuff on a display. Well,

how on earth would I make it display blocks? I had to go work with the people who were developing the display and, as I said, even the level of getting them to install a special button in the hardware so I could type my thesis out. So I think the flow tended to be much more from the research people to and from the technical people, who were building platforms.

NORBERG: Interesting question that occurs to me there, when you said going to the display people. Who were these people? What sorts of things were they doing? Were they what we would call now the operational people in a computer center? Or operational people like in my Computer Science department -

WINOGRAD: Well, no -

NORBERG: - providing all of this?

WINOGRAD: It was a mixture. Two roles were conflated in those days, which is operational people and technology researchers. So they were always building new gadgets, but they were also the ones that had to keep the old gadgets running. You went to them if it wasn't running. There wasn't this distinction - you know - somebody comes up with ideas and somebody else just keeps it running. So they were writing the time-sharing system; they were writing the text editor; they were always tacking on new pieces of hardware, trying to write interfaces that would run them. You know, they were the ones who had the job of getting the computer to run the arms, for example. Somebody had to figure out how to actually write a piece of program that waved the physical thing around, displays, that kind of stuff.

NORBERG: Would the displays be classified as graphics research at that time, by any chance?

WINOGRAD: Oh, I don't - we had two displays, one of which was heavily used and one of which was not. The heavily-used was the display that came built in with the PDP-6.

NORBERG: Yes.

WINOGRAD: It had a display console, which in that day was a novelty. They wrote debuggers and things which let you use the display console as a way of debugging. Then I did the blocks pictures on it. This guy was doing an infant simulator on [?] and things. Minsky was - there were a lot of people *playing* with display. There was something called the Minsky-tron.

What Minsky discovered playing around is that if you took two registers and you wired their contents to the X and Y coordinates of the display, and then wrote interesting little programs that did shifts and rotates and masks of the bits, you got interesting pictures on the display. And, you know, you could play around. You could change the way it did the shifts and masks and all of a sudden the patterns would change. There was no computer science theory at all, right, it was a toy. And also you could wire some of the other registers to the speaker. And I remember this thing called a Minsky-tron. It was a particular vision. I think the way it started, is he was trying to come up with a clever algorithm to draw circles on a display, to actually plot points in a circle. But it turned out that due to a bug or early attempt or something, it didn't plot circles; it plotted all this weird stuff. I remember a lot of display stuff was that kind of thing. People just sort of playing around. There was also the start of the "Space Wars"; that was on it's way. Then there was a color display, which they had gotten, which was very crude. Basically, you had to do very slow, point-by-point stuff. So even drawing a line was a huge operation. And I originally wanted to use that. I wanted to use the color display for my stuff on blocks. But it was just too hard to program.

The one program which used to run on there regularly was called "Wire Forks," which basically was, you know, you'd see this little dot going up and then it would explode into a bunch of colored dots which would gradually settle - it is now called a screen saver. And there was one guy, Mike Beeler, who was the only one who really worked on the color display. But the color display never got beyond very simple toys. The DEC display was used. Nobody was doing anything say, like, Sutherland, or something, which was really graphics oriented.

NORBERG: I have a reason for saying that, but I won't interrupt this now. Maybe later I'll tell you what I mean. Two

things occur to me in response to that story. One is, are a number of the ideas in the displays and the Minsky-tron the basis for later, say, paint systems that were developed, and so on?

WINOGRAD: No. There was a big gap there which I encountered when I came out here in '72 to Xerox-PARC for the summer, which was when it was just forming. I had been working with Papert on LOGO, and one of the things that I remember - I gave a talk in 1970 that included these huge slides of wonderful patterns that we'd produced by a program in LOGO. The fascinating thing is you could give a very simple algorithmic procedure that generated what looked like interesting and complex patterns, but that all came from this one little algorithm. I think that - that was an ideology, right? That all this wonderful and mysterious-looking stuff we call intelligence came from just getting that right little algorithm to spin it out. And I came out here and Alan Kay was working on the ancestor of all that stuff. I remember seeing this program which everybody seemed to be really happy about where you could paint on the screen. And I said, "Well, why would you want to paint on it? You can do that on paper." Right? That is, what's interesting about a computer is that it can generate things from algorithms. The fact that you can duplicate what you can do with something ordinary like paint seems totally a waste of the computing resources. And that was certainly the attitude there. That is, the interesting thing was - I mean, the game of "Life." That was another thing that was going on that the display was being used for all the time, was Gosper running "Life" patterns. The fascination was how you could come up with some interesting, simple algorithm which produced complex behavior. Not how you could use the computer as a medium for things in which the complexity lay in what the people were doing themselves. I think that was a real jump, in some sense, from that whole culture to the PARC culture which was in the other direction.

NORBERG: And the second response I have is that it occurred to me just now when you were describing the need for a color display in your own work, how did the machine know which block was green and which was blue and which was red, and so on?

WINOGRAD: Well, two questions . . . the machine knew it because it said it in the database. Real simple. There was a property on the property list of the block called "color," and this block said "color-green" and this block said

"color-red," right. How do the *people* know. *That's* an interesting problem.

NORBERG: Well, forget the people for a minute, but how did the machine know which actual diagram in a picture -

WINOGRAD: The machine wasn't reading pictures, the machine was generating pictures -

NORBERG: Yes.

WINOGRAD: - from a database. The machine had a database which said, "There's a block of size such-and-such at location such-and-such of color such-and-such." A pyramid at size such-and-such. The fact that that had anything to do -

NORBERG: The location I was looking for -

WINOGRAD: Well, there was a database. And for each block there was an entry which had size, color, location, shape -

NORBERG: Which I assume would be continually updated if the block was moved -

WINOGRAD: And when you moved it it updated the location. Exactly.

NORBERG: Okay.

WINOGRAD: So you created the scene in the database, and it was the job of the display program to put it on the screen. What it did was in neat little letters at the bottom of each block said what color it was.

NORBERG: Green.

WINOGRAD: That's how. So, what "green" meant, as far as the machine was concerned is the kind of block on which you write G-R-E-E-N in the corner. It had nothing to do with visible light.

NORBERG: Okay, so there was a level of assistance which had to do with the specialties of other people, let's say. The displays. There were various kinds of printing, and so on.

WINOGRAD: And there was the general kind of assistance you get just by the fact you're in a community of people who were talking about what they're doing. So you see a connection, but it was very...That was the sort of *ad hoc* community of processing.

NORBERG: But wouldn't we assume that that's true of most research fields, if not all? You'd expect that sort of thing to go on.

WINOGRAD: Yes. Any research community.

NORBERG: Okay. So, you then finished your dissertation and stayed on at MIT in, what I understand is the Electrical Engineering department.

WINOGRAD: Yes. The division between electrical engineering and math was irrelevant. There was a matrix structure. So one dimension of the matrix is schools and departments and so on; the other is research institutes. Papert happened to be in math, and Minsky happened to be in electrical engineering, but it was irrelevant. So, I got my degree in the math department, because I came in the math department, because I had no engineering background whatsoever in my liberal arts education. And then I switched to Electrical Engineering, because they happened to have a slot available that Minsky had control over.

NORBERG: That *Minsky* had control over? This was not a faculty decision about who the faculty would hire?

WINOGRAD: Oh, when I say Minsky had control, I mean in the political process sense. I don't mean, necessarily, that the school said, Minsky you fill the slot. But, you know...

NORBERG: Well, it happens.

WINOGRAD: I honestly don't know. It's true with the funding, too. This was the milk-and-honey days. You didn't have to think about that; Minsky took care of that. If you needed slots, he got them. If you needed money, he got it. You just did your work. Junior people didn't think about those issues. So, I have no idea. I mean, I said, "Well, I'm done with my thesis, I'd like to stay on." And he said, "Okay, we'll make you an instructor."

NORBERG: And did you then continue on with the research you had been doing, or did you begin then...

WINOGRAD: Actually, when I got my degree, I had a very rudimentary program in rough write-up. In the course of the next couple of years after that, I re-wrote the dissertation to some extent, working with Walter Reitman, who was the editor of *Cognitive Psychology*. That became the book, so a fair amount of time went into re-writing; a fair amount of time went into actually reprogramming the system. Stu Card came up from Carnegie for awhile. We had a couple of students there, and we re-software-engineered it, is the best way to say it. All my data structures that I had written were implicit in list structure. So I knew that to get the semantic features from something you right, and we actually converted those all to structures with names, and some documentation. The kind of apparatus that makes a program more readable and useable. And, I was thinking about the problem of ambiguity in context, and come up with ideas which are in the frames direction. The kind of stuff that Minsky wrote in the frames paper. Again, no one person was working on that. That was sort of a set of issues that were being talked about; how do you deal with larger structures. When I wrote about this in the dissertation, I said one of its limitations is that it's moment-to-moment; it only deals with a particular situation. The example I used is: Jimmy came into the room carrying a present and Billy opened the box, or something. How do you know? Well, it's because you have a frame for birthday. I would now say, you know, you have a frame for birthday presents. How do you integrate that? And that was all. I

honestly don't remember doing new programming technical kinds of things during those couple of years I was there. I think most of what I did was cleaning up the program that existed, and writing about it, and writing speculative papers and thinking about this sort of frame kind of idea, but not trying to implement them. Then I came out here in '72 to work with Bobrow.

NORBERG: Why would it be necessary to clean it up?

WINOGRAD: Well, because I think the feeling was that...Certainly, there was feeling that other people would want to use it. Which other people, I don't know. There was certainly no sense of making a profit. It wasn't a sense that it would be a product. But that it should be a platform on which to build. That I had handled a certain sub-class of the problems that needed to be dealt with, and somebody would then add to it things which would handle more of the problem. So therefore if somebody is going to add to it, it has to be in a form which other people can understand well enough to actually do things with it.

NORBERG: Let's get back to the incidental comment that you made about coming out here to Xerox-PARC and meeting Alan Kay and others just as Xerox-PARC was getting going. When was that again?

WINOGRAD: The summer of '72.

NORBERG: The summer of '72. After that brief period, did you go back to MIT?

WINOGRAD: Yes. That was a summer job. I had gotten to know Danny Bobrow when he was at BBN. And he was one of the people they were bringing out to the PARC, and his job - I think he just came in '72 - was to create a group in natural language. So, there was an obvious reason for him to want me to come out and work for the summer.

NORBERG: Sure. A group in natural language at Xerox-PARC, not at SRI?

WINOGRAD: No, no. Xerox-PARC. This was one of the major groups in the initial Xerox- PARC. Bobrow and natural language was a major chunk. One-fifth, one-sixth, I can't recall. There was Alan Kay's stuff; there was the Bill English group doing office systems; there was Butler Lampson and his people doing, basically, advanced architecture; and there was the AI group, which was a natural language group, or that was AI.

NORBERG: Now, would that have been similar to the natural language group that had been at BBN for several years, supported by DARPA?

WINOGRAD: I can't give you a quick answer to that. Certainly, historically the answer is it had a lot of connections. Because not only did Danny come, but he brought Warren Teitelman, who had been, not natural language, but doing the LISP stuff there.

NORBERG: Yes.

WINOGRAD: So, there was a continuity there. I think the charter was much more specific to do natural language systems that Xerox could sell in some sense. I sense the BBN group was like the one at SRI and so on. It depends upon what DARPA wants at the moment. So you were contract seeking. Where at Xerox, there was more of a sense of we're going to build a particular kind of product, and you don't go off and look at whatever happens to be fundable at the moment. And the product was going to be basically question-answer. That was the charter - Ron Kaplan. Martin Kay was a little bit later [?] fairly early group. And the idea was to be able to have information systems, which included natural language.

NORBERG: Okay. Good. Then you returned to MIT, continued with the activity -

WINOGRAD: So then I went back for - I'm not thinking - I may be confusing years. Because I know I went to Carnegie in the summer of '72; maybe I did both. I think I did both.

NORBERG: Well, the next question would be when did you come here and how did that occur.

WINOGRAD: Anyway, I came to Stanford with very simple logic, which is my wife had just finished medical school, and was applying for internships. And we went around the country looking - we inquired around the country looking for places that had an internship in family practice that was good, that she wanted to do, and a university that did AI. And there weren't that many good programs for her, that she liked. In fact, one that we actually traveled to was in Florida, in Miami. It wasn't clear *what* I would do in Miami [laughs]. I went and talked to people there, Institute for Studies there, and stuff. It turned out that San Francisco was a program that she liked a lot. So, I had come out and given a talk at Stanford earlier, on my dissertation stuff, and I basically said, "Are there any possibilities?" It turned out that Roger Schank was going on leave for a year to go to Yale in '73, so they said, "Well, that's nice. Why don't you come out and be an acting assistant professor as Roger Schank's replacement for a year while he goes to Yale and then we'll see." And Roger Schank never came back. So when he didn't come back, basically his slot was open. I was already here, filling it, so I just took it.

NORBERG: Well, just incidentally, why not telephone Xerox-PARC and say, "Hey, we want to come out to San Francisco for awhile"?

WINOGRAD: Well, that was a question. During my first few years here, I was a good fifty percent of my time at the PARC. I was a consultant, and I was using facilities they had there which were for things just like word processing, which were better than what I had at Stanford. I wanted to be in academics; I like teaching. Of the models of being in a research lab and being in a university, I like universities. I wanted to do *both*, which I was able to do, so it was a good deal. It worked out perfectly, right, to be at Stanford, to be connected with Xerox, to be where Carol had her internship.

NORBERG: Yes. I know that well.

TAPE 2/SIDE 1

NORBERG: When one is taking the position of someone else who is on leave, is it possible to continue an established research program or not?

WINOGRAD: No.

NORBERG: Did you feel sort of uneasy during that year?

WINOGRAD: I didn't feel uneasy, because I just did something different, which is I, basically for the first, I would say, three to five years I was here - I would have to go back and try to think - did my teaching at Stanford and my research at Xerox. I simply made no attempt to create a research program or to fill in for Schank's research program. But I would be at Stanford, teach courses, talk to his students - and I sort of - I became a *de facto* substitute advisor for a couple of them, and it worked not so well. You know, they finished and got things done, but it wasn't like I really got that involved with what they were doing. And my research was all in connection with the group at PARC.

NORBERG: It's my understanding that there was very little contact between the Stanford Artificial Intelligence Laboratory and the people in the Stanford Computer Science Department. Do you remember that as having been the case when you arrived in '73, or not?

WINOGRAD: There were three groups here. One of which was so far out nobody ever talked to them, which was the Stanford Computer Center. Those were the people who actually had the mainframe and ran it. They had had a huge feud with the Computer Science Department over whether to use IBM equipment or other equipment, and so on. All that happened mostly before I got here, so I only saw the fallout from it. So that was, you know, the third force. They were really completely separate. Then there was the Computer Science Department which, at that time, I guess we were in Poyla Hall. I've forgotten. And then there was the AI lab, which was five miles away. Basically people in the AI lab came down to campus when we taught courses and, you know, if there was a faculty meeting once a quarter, you came to that, or something. But other than that, the community was the AI lab. There wasn't animosity.

There was animosity with the Computer Center. There was animosity at MIT between groups. Here it was just that they were doing other things. They were doing algorithms, or whatever - it's not our stuff, right, we're doing AI. And the AI lab was really a community unto itself.

NORBERG: Did you affiliate with the AI lab?

WINOGRAD: Yes. My office was there. I didn't get an office on campus until several years afterwards, three or four or whatever, after I got here. It was just too much trouble to try to teach and be up there all the time.

NORBERG: Can you make a comparison between the AI lab at Stanford and the AI lab at MIT for me? In terms of the camaraderie of the people, the kinds of research problems, results maybe?

WINOGRAD: It's an interesting question to try to think back on. The MIT lab had a certain kind of intellectual focus, driven by Minsky in his own chaotic way. He's not an organized person, but he had certain kinds of ideas that he really cared about and pushed other people to work on and would interact with people working on them. There was much more of a sense that the project you picked and how you would do it was to please Minsky, even though he'd only come and dabble in it. He wouldn't be involved actively. I think McCarthy is much more of an individualist, and I think the feeling of the AI lab here was do your own thing, and find other people who want to do it with you, but it's not what John's interested in. He's sort of irrelevant. He goes out and gets the money. So, I think it was less coherent. It was less of something I would call the AI lab way of thinking or style. There was Schank and he had his bunch of people and he was doing his kind of thing. There were the mobile robot people, and they were sending carts out to roam around and doing their kind of thing. And there were the hand-eye people, and they were doing their kind of thing. And Binford was doing his vision thing. But there wasn't much of a sense of coherence among them. And of course, John, who was the leader of this place, he was interested in logic, so he had people doing logic. But much more sort of each sub-group just doing its own thing and not particularly interacting together. There was a shared facility, everybody had to interact to some degree around SAIL and developments in SAIL.

NORBERG: Why did they have to?

WINOGRAD: Because it was by far the most powerful computer available to those people.

NORBERG: I guess I'm misunderstanding something then.

WINOGRAD: SAIL meaning the computer, a time-sharing system.

NORBERG: As opposed to the language, as opposed to the name of the laboratory. All three of these things can apply.

WINOGRAD: Right. SAIL is the name of the laboratory; it was a place. SAIL as the name of a language, very few people used. SAIL as a time-sharing system was the thing, as in many of these early labs, the thing that held everybody together was they all had to deal with this common time-sharing system. Everybody knew - you know, had to deal with other people - how to use the text editor, and what happens if my screen doesn't map. Actually, there's an interesting idea here. One of the ideas that this time-sharing system had that, only today with Timbuktu and sort of advanced hardware, advanced software things you get back, was that you could look at anybody else's screen. It was run through a video map. And there are a lot of technology issues there, but you could be sitting in the lab and sort of browse around to see what was on other people's screens without leaving your chair, right? You also could block, but that was considered impolite.

NORBERG: Now why would you want to look at other people's screens? What's the point of that?

WINOGRAD: Well, the *claim* was, for example, you could work together. Somebody's having trouble, they can call you up and you can say, "Let me see your screen," right?

NORBERG: I see.

WINOGRAD: If you're working on a vision project, you could have the camera image on your screen and somebody else could be working on their screen and look at the camera image. It was an accident of technology. The accident of technology was because display hardware was expensive. They had a small number of display controllers and a large number of video terminals and a video switch. So, somewhere in one central place were all the actual display controllers that generated the common displays - and then all you had out in your office was a video switch which said, Okay, map me on to display #43. So, the ability to map on to anybody's display was a natural product of that. I mean, it was just as easy as mapping on to yours.

NORBERG: Yes.

WINOGRAD: In terms of the people around, I guess it was less hacker-driven. I think the MIT, the sort of Minsky's love of sort of playing with the mechanisms and people like Greenblatt and Knight and that group, was a core that everybody else related to. And here there was more of a sense that the people who did SAIL were a group, but they weren't...You wouldn't naturally go and talk to them or be involved with them unless you were part of that. The language people did their own stuff in language. It was sort of separate.

NORBERG: When I read the proposals to DARPA from the two organizations in the period, say, '65 to '75, which is one that I'm concentrating on very heavily, there is an attempt to show coherence on both sides, and it appears to me as if the MIT people were able to pull it off, and the Stanford people are not - on the one hand. On the other hand, there are similar areas: there's the hand-eye project going on here, as well as the one going on at MIT; there are claims in MIT proposals that Planner and MicroPlanner are programs that are being shared between MIT and Stanford, and so on. That's before you came here, by the way. I'm not trying to suggest that there's an obvious connection there; there would be afterwards. But, I'm fairly confident that it was '71 that the proposal stated that MicroPlanner was in the process of being developed, and the concepts in Planner were being shared with Stanford. I think it was '71; it could have been '72.

WINOGRAD: I don't think MicroPlanner ever ran here, because it ran on the Maclisp; Maclisp didn't run on SAIL.

NORBERG: I think the point of the MIT proposal was that "See, we're developing things that other people are using."

WINOGRAD: Right.

NORBERG: I don't get any of those statements, or at least I can't think of any of those statements that would have come out of Stanford proposals. Now, getting back to your statement, though, which I think is consistent with what I just said, that there is less, I won't say cooperation, I don't think that's right, but there's less *overlap* among the research groups at Stanford. There is a sense in the documents that I've seen, particularly the historical documents done at the latter part of the '70s, that Les Ernest was sort of keeping things in some sort of cohesive fashion moving forward, keeping things moving forward in some sort of cohesive fashion. Do you have the same feeling?

WINOGRAD: Well, Les had two different roles. The first role he had was he was the Manager. John McCarthy is many things, but being a manager is *not* one of them. And he just had no interest, inclination or ability to manage a group. So everything having to do with making things run, and finances, and all that kind of stuff - publications - all that kind of stuff, Les was the person who was really doing it. So, in that sense, he had a lot of central - not *power* so much as - power in the sense that anybody who runs things has power, whether they have the official policy-making power or not. He was also the sort of - I don't know how I want to say it - a role that Minsky had more - you see, McCarthy never was a "hacker." McCarthy is in logic and mathematics. He's not the kind of person I would see sitting down doing an interesting graphics program that draws neat little circles; it's just not his style. Les is much more like that, so that particular role which Minsky played in addition to his sort of lab director role of coming in and saying, "Wow, that looks neat! Let's develop it." I mean that kind of thing, I think Les played much more. It was true in the development of the time-sharing system. I think it was true in just various things.

NORBERG: So there is such a guiding light behind the organization.

WINOGRAD: To some extent, yes, although he was - I think he was - he picked up the sort of hacking end of it. Minsky also had, of course, a view of AI. I don't think Les had such a view; he's always been more of a gadgets person. I think Les had a lot to do with the social environment, setting the social environment - the sauna in the basement and, you know, these kinds of things were more his style. John just didn't care; he is not a social creature.

NORBERG: Okay. All right, now, while you were here you began working with Bobrow and others on KRL. What was the genesis of that project?

WINOGRAD: The genesis was we wanted to build a natural language understander. It was clear that the representations that you could do in something like Planner or MicroPlanner or so on were limited in very serious ways relative to what you needed for dealing with context and ambiguity and all these bigger problems in language. We wanted something which had some of these frame ideas, and the general - again, Xerox was in a tool-building phase. The idea was there are wonderful applications out there in the future and we are pioneering wherever we can do it. Building underlying tools - you know, time-sharing systems. They were going to build a time-sharing system . . . they did, actually. What was it called? MAX. Then they built the ALTO. So, I think the spirit was that the tool that we're going to need for doing natural language is a more sophisticated representation and inferencing system, which has some of the psychological properties - that was the thing that differentiated it, I think, from a lot of the previous things. We were consciously thinking about the fact that human inference and human memory have limitations and properties that aren't the same as pure logic. There's the notion of a fact being available but not easy to recall.

NORBERG: Yes.

WINOGRAD: So it was motivated by some *melange*. I now, in hindsight, think it was hugely naive to try to pull this together. Psychological considerations about memory; that came out of the semantic net literature - stuff like, what's his name? Quillian?

NORBERG: Quillian.

WINOGRAD: Representation languages that you could process, which came out of Planner and that kind of stuff. Logical semantics related to representations, which came out of more of the McCarthy style work. And readability and understandability, which came from sort of software engineering motivations. I mean, a representation language had to do all sorts of things. One of the things it had to do was be something that you as a knowledge engineer could read and write. So KRL was an attempt to produce a language which was nice to read and write for the engineers who had to write programs in it, processed like human memory, so you could have realistic AI programs, had an underlying semantics which was firmly grounded like logic languages, all in one, all in one language. And I think it - again, in hindsight - it just bogged down under the weight of trying to satisfy all those things at once. In addition to the inherent difficulty. Other AI languages have solved problems by avoiding some of those.

NORBERG: What was, then, the influence of KRL?

WINOGRAD: The influence of KRL was, I think, a couple of things. One is it was the first language to put this sort of frame-like style into the heart of a representation language. And that got picked up. They wrote this thing called FRL, Frame Representation Language, at MIT. And the various "RLs" following that, I think, picked up on that idea. I wouldn't say the idea was brand new when KRL was in the air and all this stuff about frames, and so on, but it was the first concrete embodiment. I think the other was to make respectable the notion of a coherent representational language that was neither an *ad hoc* mechanism like the semantic nets nor a representation of formal logic, like the SRI stuff and the McCarthy stuff. So there's something which you can actually talk about as having a coherent, formal structure beyond just all the data manipulations. Property lists are a representation language, but they don't have any semantics. So you could have a semantics of a language which, nevertheless, didn't have the standard properties of, say, first-order logic. I think that general attitude towards what a representation language could be was at the heart of a lot of that stuff, which is, you know, center question mark and sort of representation language stuff. And it wasn't so much any particular technical device as a sort of sense of what a representation language was and

how it fit into the AI system.

NORBERG: Now, I haven't followed the representation language development. So can you tell me whether or not work continued on KRL and various other kinds of similar languages - whether that kind of work continued at Xerox-PARC, whether it continued here, or went elsewhere?

WINOGRAD: Okay. It did not continue at Xerox-PARC. We're jumping ahead in the story, but as of about the late-'70s, I lost steam on that project. Bobrow got diverted into building - trying to get LISP onto an ALTO, which we saw as a necessary sub-part for moving along. It was really the two of us who were pushing it. Other people in his group were doing more of the traditional linguistic side of natural language. Kaplan, for example, who's still doing syntax and parsing and stuff. It never caught on at Stanford; there was never any work of that kind here, in those days. It got picked up later by, for example, Mark Stefick. He did - what do they call his? - he had a knowledge representation language. I want to say "Objects" - I'm not sure. And sort of got picked up, to some extent, in Ed Feigenbaum's group, the heuristic programming group, but not directly; it wasn't an ancestor of that, it was just doing work of that general kind. It got developed much more at BBN, for example. Bill Woods, Sam Brachman and those people set up a project to do KLONE. Rich Fikes at SRI did some of that. But none of them were descendants in the direct sense that a body of code moved along and got developed. It was much more in a loose sense of that kind of enterprise got developed.

NORBERG: Can any similar statement be made about SHRDLU that certain blocks of code were transferred to other programs later on by others?

WINOGRAD: No. SHRDLU, well, except there were *minor* cases. One of my students at MIT went off to CCA and wrote a program, basically taking SHRDLU, which could answer questions about the weather. But it wasn't, you know, it wasn't a line of development. It was basically an exercise in changing the domain of SHRDLU. As far as I know the code - I wrote a fairly detailed description of the workings in the book, so a lot of things which people went ahead and did I'm sure were influenced by the fact it said, "Here, you can do the structure this way and this way and

this way." But as far as actually picking up pieces of LISP code - well, I'll give you the obvious answer which is it ran in Maclisp. So nobody who ran anything except ITS could run the code. It never got imported to any other dialect of LISP as a whole. Now, maybe somebody - you know, I got - I've had requests over the years, "Please send me your source code." And I have no idea . . . maybe some imported version is running in a Mac in Timbuktu, right, I just don't know. But as far as a project that had major visibility within AI, everybody started from scratch and went their own way.

NORBERG: Well, let me ask the question differently, and try and get a similar kind of answer in a larger sense. If the blocks world was essentially part of the *milieu* around the MIT AI laboratory...I was looking at one of the exams - I don't know if it was exam questions or homework assignments in one of the courses taught here at Stanford - not yours, but one of the other professor's . . . Generith?

WINOGRAD: Genesereth.

NORBERG: - Genesereth, using the blocks world in one of the exercises that had to be done by the students, suggests that the blocks world is still very much around, and no longer has to be part of the MIT environment. How did that spread occur? Do you know the history of that at all?

WINOGRAD: Well, certainly the blocks world was at Stanford. McCarthy came here from MIT and set up a research lab for robotics, which was very much like the one at MIT with the hand-eye coordination and stuff. So, when I came here they were moving blocks around on tabletops.

NORBERG: Well, the chronology there might be a bit of a problem, wouldn't it? He came here in '61, or '62 depending upon how you count the year off, and as far as I know the hand-eye projects were later than that in both institutions.

WINOGRAD: That's all before me. I didn't get to MIT until '67.

NORBERG: And also, when I think of the Roberts work, which is '63, in which the blocks world gets sort of defined, McCarthy would have had to import it.

WINOGRAD: Well, he imported it. You say it more generically, because you know the history and I don't, in detail. But the community of discussion about AI - and I'm sure the DARPA contractor's meetings were a big piece of that, and so on - the blocks world was just in the air.

NORBERG: Okay. And it's in everybody's air, I take it.

WINOGRAD: In everybody's air, right. I remember people were looking for the *Drosophila* of artificial intelligence. You want to find the thing which is easy to work with, but which gets you the interesting issues, as opposed to the messy problems that you get into if you try to tackle "real world" things. So, the vision projects - all the vision projects - well, not *all*, because there were probably some sort of aerial photographs, but the ones that were in that particular milieu all dealt with nice geometrical objects, because that's easier to run vision programs on. I think the blocks world really came more from the vision side more than the manipulation side. The ease to recognize objects was the first thing.

NORBERG: Yes. Coming out of pattern recognition. And when it gets defined as a CS problem, then we see the origins of the blocks world. As far as I know, that's out of Roberts' work. I've deflected you a little bit in trying to appreciate something, and you just clarified something for me that I wasn't aware of, and I'll have to now go back and rewrite a section because, in effect, I had the blocks world being generated at MIT, and starting with Roberts and then going to your work, and following on from there. Maybe that's a bit of an exaggeration on my part, so I'll have to look at that again. All right. Returning now to Stanford - we left that when we went to these incidental areas - you commented on the activities going on in the AI lab, and you commented a little about McCarthy and the funding, and so on. Who else was there at the time? And I'm particularly interested here in the comparison, again, with MIT. There you were able to note all of the - not all, but a number of the graduate students that you were, I presume, both socially and professionally related to, and so on. Here you might have a different perspective as a faculty member. Can you try to reflect on that in the same way?

WINOGRAD: Yes. Let me take off pieces of it. There was Schank, and, of course, he wasn't here when I was here, but his students were. He had quite a few, six or seven students, some of whom followed him to Yale and others stayed here, and some never finished. It was a complicated situation. But he had a fairly big enterprise, by those standards, going, which was unto itself. It wasn't connected to anything else going on; it was his scripts - well, it wasn't scripts in those days, it was conceptual dependency. Then there was Colby, who had his artificial paranoia thing, and he had several people working for him, including Larry Tesler, and I've forgotten who else, Roger Parkinson. And, again, that was pretty much an enterprise unto itself. Those were in natural language, so those were the ones I knew most directly. Then in robotics, there were the cart people. They were doing mobile robots, and I think Moravec was involved in that part of it. And then there were the hand-eye people. Vision. Binford had a group doing basically object-finding, that kind of stuff. There were some vision people working with the cart people, trying to deal with the problem of visioning - of, you know, driving a cart around. I don't remember who. I remember they used to make great use of these display screens for interesting visually processed pictures of the thing the cart was seeing on this TV screen. And then there was the hand, the actual, physical hand, and they had this *massive* industrial arm, and it used to just shake the building when it stopped and started suddenly. It was bolted down to the concrete. I don't remember who was working on that. Then there was Jerry Feldman, who I honestly . . . I mean, he was working on the SAIL language and I can't remember - he had other research projects. I think he was doing stuff on vision, but I can't remember exactly what he was doing. He left fairly soon after I came here, in the next year or two - I've forgotten just when. I don't remember any - McCarthy was there, and he was doing his logic stuff, but I can't remember a group. There must have been other people working with him.

NORBERG: There were a few, but not many at a time.

WINOGRAD: Yes, sort of - and they seemed to be more on their own. There wasn't a sense of . . .

NORBERG: Involvement?

WINOGRAD: The music people. We shared the building with the computer music people. I know Leland Smith was there doing his score printing and stuff, and John Chowning was doing some of his early stuff. A certain amount of the social environment of the lab was very different from MIT, because they brought in these people from the arts world. Very different characters than you'd find on the ninth floor of Tech Square. Plus, it being California.

NORBERG: You do have a sense of all these groups sort of working independently?

WINOGRAD: Yes.

NORBERG: Nothing like what you had observed earlier at MIT?

WINOGRAD: Yes. MIT was only *loosely* joined, but this was more little separate groups.

NORBERG: Do you think that the graduate students in SAIL worked in the same way as the graduate students at MIT had worked, or had the field gotten somewhat more sophisticated by that time? We're talking about almost the middle-'70s now. The same sort of intuitive approach that you described earlier.

WINOGRAD: Certainly through the '70s.

NORBERG: How about connections with SRI?

WINOGRAD: I don't remember much. There was an open discussion. I mean, certainly I knew what was going on at SRI to some extent. They had a language group. There was SHAKEY and so on, but there wasn't - there's been a pattern at Stanford, which still exists, which is a certain number of graduate students get farmed out. So, there are graduate students at Stanford who are really graduate students at SRI, or graduate students at Xerox-PARC, which means that there's a certain back-and-forth flow. They really detach from Stanford in terms of their social and day-to-day work to do that, so it's not like the group pulled together, but there's a certain permeability because of students

going back and forth. There were certainly students working at SRI and students working at Xerox-PARC in that way.

NORBERG: Why wouldn't that have happened at MIT, with Lincoln Laboratories and BBN, maybe Draper.

WINOGRAD: Draper was too different in its approach. BBN would have been a logical place. I don't know. That would be in the domain of politics and funding, and so on, as to why those places didn't. I know Lincoln did, to some extent. In the early days they did . . . people like Roberts and Sutherland, and stuff, were connected with Lincoln. By the time I was there, there weren't students at Lincoln, and I don't know, it may have just been the difficulty of distance. I mean, it was not close and convenient. BBN did it a little bit. I'm trying to think of a student - I don't think they had the funding or the idea or whatever . . .

TAPE 2/SIDE 2

NORBERG: You stated a few moments ago that after you and Bobrow had worked on KRL for awhile, toward the middle-'70s you - or, I guess that would be the later-'70s, wouldn't it? - you began to drift into other areas. What were they?

WINOGRAD: Well, the immediate history was, in the middle of working on that - and at points we were hitting some technical snags having to do with the underlying implementation language and stuff, and therefore there was a good argument for waiting until some things, you know, a new LISP system got built, and so on. I was also teaching, and I decided I wanted to make my mark in the world by writing the definitive text. So, I set out to write the definitive text on natural language processing. And that diverted huge amounts of my energy. Basically, the research energy got turned into book-writing energy. And I wrote what was going to be one book, and then it got bigger, and bigger, and bigger and I decided it was going to be volumes. And it finally came out in '83, I guess.

NORBERG: Which one?

WINOGRAD: *Language as a Cognitive Process.*

NORBERG: Yes, '83.

WINOGRAD: And I just, you know, I got into that naively in two ways. One is not knowing how much time it takes to write a book. It's just a big job. But the other is believing that somehow I could write a book about natural language processing that was introductory - people could read it without a lot of background - which was comprehensive, so once you'd read it you'd really know what was going on in the field, and which was innovative in the sense that it said "here is the right way to do things." So, it was going to be an introductory text, an encyclopedia and a research monograph, all wrapped in one grand synthesis.

NORBERG: Which part got dropped out?

WINOGRAD: Well, the answer is: for syntax, which is the part I actually got written, I dropped out the monograph part. See, syntax isn't my area. I'm interested in representation. So, I can go ahead and be elementary and encyclopedic, basically. And I would say, for what was going on at that time, it achieved a reasonable coverage that way. I think it was successful in those. You don't read that - I mean, people used to read it and say, "Well, where's *your* view of what really should be done here? All you do is describe all these possibilities." That's what I did; I didn't push it farther. And, on the semantics I couldn't do that because I did care, but the things I cared about I couldn't - I mean, it got longer and longer. But it became clear in an implicit way, gradually, that I was never going to get a book to do all the things I wanted. And by that time I'd gone along far enough to ask the question, "Well, is it worth the amount of time it would take to do one of those things well? Is that what I want to spend my time on?" And Volume II never happened. I have the book on my vitae which says "Volume I" and that's all. But I think my energy really got diverted from research in the KRL area to writing a book about language for a period of five years or so, if you take the amount of time I was really working on the syntax part and then the time I tried to develop the semantics part, and so on. During that time was when I also started having more of these conversations with people

who were skeptical about AI. So, in the early to mid-'70s - I should go back and pin down the date on this, I don't remember it - somebody - I think it was Bert Dreyfus, but I'm not sure - initiated a lunch seminar. Very informal, in Berkeley, where he and John Searle and various students of his came, and Danny Bobrow and I and various students of mine - and it was like once a month we'd just get together and talk, and so on. Then, in the midst of that - or the end of that, I've forgotten the exact sequence now - Flores, Fernando Flores ended up at Stanford. And that's a whole other history. His history. But I started talking to him and what's clear, in hindsight, is that a lot of the sort of doubts and difficulties with the AI paradigm that I was feeling I had already been - they were stewing - I was ripe in some sense. I had run up against a particular problem that it was clear that I didn't see a few more steps that way solving. KRL had reached a certain level of complexity, and it wasn't - I think at some intuitive level - I wouldn't have said this in those days - I could see that it was going to bog down in its own complexity before it solved the problems in control language. And that partly has to do with this business, as I said earlier, trying to do everything at once; to be a representation language and a programming language, and so on. But I was certainly feeling at some level - and I think trying to do the book on semantics - again, just realizing the more and more complexity of all the different issues that were coming in without a unifying feeling that they were coming together. So then, in the course of I would say '76 to '80, more or less, I went through this gradual shift away from saying well, "I've got to get back to getting KRL to work," which was the feeling, "Once I get done with my book, I'll go do that," to feeling, "Well, once I get done with my book on syntax, I'm not sure that I want to pursue AI in the same way. I'm not sure that's the direction to go." And in the course of that, I started working on the book with Flores. And the earliest - I don't know when the earliest draft was about - late-'70s that we actually - I think we originally promised the publisher an '81 publication date or something like that. And then I just gradually shifted communities in some sense, an interesting process of being less and less involved with Danny and with the AI people. He had gotten diverted for reasons having to do with restructuring a project at Xerox. And Bob Taylor's view about AI. That's another part of the history, which you may or may not have chosen to get into. The situation there was shifting in a way which just didn't make it convenient to continue. Not that we couldn't have. And I became more and more interested in the stuff I was doing with Flores, and the issues that that was raising and the directions and by the mid-'80s - or the early-'80s - my book was published in '86, but it was pretty much in that form earlier - I'd really come around to this philosophical questioning of AI, as opposed to "We're headed in the right direction. Let's just work harder and do more," which is

the spirit I had had at MIT.

NORBERG: Now, what are the implications of that for your research at the moment?

WINOGRAD: Well, basically my research completely shifted. I don't call myself an AI researcher, or an AI person. Officially, in the Stanford department, computer science is divided up into five chunks, one of which is called AI. I'm still in "AI," but I see my work as being really human-computer interaction, as opposed to AI. The question is how do you use your understanding of how people think, in whatever form you have it, to better interface that with computers rather than to duplicate it or model it in computers.

NORBERG: One could argue that, back in the early-'60s anyway, when people were beginning to define what became the major research problem - as far as I'm concerned it became the major research problem - for computer scientists over the next decade and a half, was the human-computer interaction. It went in various directions: there was the graphics area, there was the AI area, there was natural language programming both as part of AI and then as the automatic programming part which is outside of AI, and so on. At least the rhetoric surrounding the justification for those programs and those directions was to improve human-computer interaction, and to improve it in two ways. One, in the technical sense that the computer is a better tool. But also in the sense that the pair, the human and the machine, become a better mechanism for problem solving, for decision making, and whatever. Now that may be IPTO hype; we might conclude that. But on the other hand, that would seem to me, then, to say that you had focused narrowly, technically for awhile, but then broadened out into this larger quest.

WINOGRAD: No. I think there's a difference there. Of course, in terms of IPTO, that comes from Licklider, right? He was into human-computer interaction before he got into computers; that was his background. And I assume he plays a prominent role in the whole story, but you're right . . .

NORBERG: *Not so much*, not so much.

WINOGRAD: IPTO hype, of course, is always a question of what people think will get funded or how they should paint it so it will get funded. Certainly, within the AI work, and computer science as a whole, of course, is a much bigger question. There was a taken for granted assumption, which is if you can get computers to be more like people they'll interact better. Given that, let's not talk about interaction, let's talk about getting them to be like people. So, the focus of the thinking and the research was not all on the nature of the interaction with them. It was on making them be like people, because that was obviously going to make them good to interact with.

NORBERG: Yes.

WINOGRAD: So, you could advertise this as, of course, the reason we were doing it was because of interaction but, as a discipline, it was not looking at that interaction very much at all.

NORBERG: Is there anything wrong with that approach? . . . to say we want to make the machine more like a human?

WINOGRAD: Well, I think for most tasks it doesn't work very well. Other than that, there's nothing wrong with it. The example I use in sort of casual talk is: Imagine [that] instead of a steering wheel in your car you could talk to it . . . "Go a little bit left. No, a little bit more right. No, no *not there!*" The steering wheel turns out to be a pretty good way to go down the road, right?

NORBERG: Yes. [laughs] Point well taken.

WINOGRAD: So, if the task is - well, take drawing. Where graphics has gone. Where graphics has gone is *not* tell it in language what you want drawn, it's very sophisticated paint programs, which let you manipulate and do things but by direct manipulation. Direct manipulation is the opposite of making the machine like a person. And I think the notion that for the tasks you want you want to do it like a person, even in the more linguistic areas. Let's get away from driving and painting, take medical diagnosis, favorite topic. Always promising money for medical things, right? If you say the machine is going to be like a person, then what you'd like to do is ask it a question and have it give

you an answer like a person would. If it's not like a person, it might show you a picture - give you a model - run a simulation. There's all sorts of things it can do which are not what a person would do if you were to ask them a question. Which may be much more useful for a practicing doctor, and may be based on all sorts of knowledge and all sorts of stuff. But, in the end, you aren't trying to - the goal of saying why don't we make it let you converse with it as though it were another person, is an arbitrary decision about how it would be useful.

NORBERG: All right, but couldn't we argue that - and I'm playing the devil's advocate here, I'm not trying to portray my own opinions at all, though I may be in some instances - but couldn't we argue that the achievement of this position, that we should not be trying to make this machine to be like a person, instead we should focus on interaction, could we argue that that is the result of 30 years of trying to understand how to represent problems, how to inference conclusions, and so on?

WINOGRAD: Well, there's two different points. One is the fact that people asked now pose the question this way is not the result of those 30 years, people were saying that ever since the beginning. There were plenty of people in computer science who said from the beginning computers can do wonderful things but making them into poor imitations of people isn't the way to do it. Understanding where the *limits* are - if you can say of any technology, if it doesn't do all the things you wanted it to, in the process of learning why you will learn a lot about the limits. Certainly you could say of AI that the work that has been done has given us a much better picture of just what it does well, what it doesn't do well, what the big issues are, what . . . The expert systems field is a great example of that, where it's moving away from the sort of generic notion of, well, anything a human expert can do you can have a program do. There is now a lot of experience with what kinds of tasks expert systems do well. Some of them much better than people, for reasons that have to do with more memory, more rules, more statistical processing, whatever. Some of which they do much worse. And you're getting some finer-grained understanding of that. Without the AI work you would never have gotten to the point of being able to get that finer-grained understanding.

NORBERG: Agreed. Agreed. But now one might be able to say that it's outside the realm of AI, because the development of a particular expert system is no longer a question of research.

WINOGRAD: Now you're back to the question that you must have asked more than once, right, which is, "What is AI?" When are you inside the domain of AI and when are you outside.

NORBERG: There are lots of answers to that question, but now I'd like to hear yours.

WINOGRAD: Well, what I have generally said was that AI, as a phrase, refers to three somewhat related but different things. One of them is a quest. The goal, which started back in history, right, in Greek history, of being able to be the creator of something which had the same kind of intellect as ourselves.

NORBERG: Yes.

WINOGRAD: That's the age-old quest. Whether you're doing it with gears, or with silicon, or with neural nets, or with genetic cloning, the quest is still the quest. And I think a lot of the fascination with AI comes from that dimension of it, regardless of the technology. And people who go into AI often go into it because of that. The second thing that AI is a particular community of people who choose to identify themselves as being in a discipline - it's a Kuhnian view. AI is what AI calls itself.

NORBERG: Yes.

WINOGRAD: Questions like, "Is neural nets AI or not?" have to be answered in that dimension. If they don't go to the AI meetings anymore and they don't publish in the AI magazine, they aren't AI by that definition, even though they're engaged in the same quest. There's a social entity which goes by that name. The third is a particular body of techniques which have been the core of AI for the last 30 or 40 years, which includes deduction, search, symbolic representation, the stuff which we ask students on the qualifying examination. Now, in some sense you could choose to believe that the people who call themselves AI are the ones who use those technologies and are the ones who are going to come up with the true answer to the quest. That's what we all believed back in the AI lab in

Nineteen-sixty-whatever, right, and therefore there was no problem. What's happened now is, of course, that those are all different. So, the technologies, which have been developed, are good for very limited tasks. They're good technologies, just like linear programming is a good technology. There are a set of algorithms, a set of programming techniques which, for certain tasks, work well. If you say, "Are they going to give you the answer to how the mind really works?" - no. They're not going to do that. There is a particular fight over who gets the name "AI," and I see that here when people like Feigenbaum and McCarthy are vehement that Rumelhart is not doing AI. Whatever it is he's doing, it's not AI. And he doesn't even have an appointment in this department. And then there's the question of, in the grand future, what's going to be the answer to the quest? Are we going to be able to build the model of ourselves? That may turn out not to be a straight-forward extension of either what is the technology of AI or what's going on in neural nets, but you know, it's a possibility. Somebody could build something. It may come out of genetic engineering.

NORBERG: Okay. Can we assume that the quest has not changed over the last 30, 40 years?

WINOGRAD: Well, the quest exists, but not all of the people who call themselves AI have that quest.

NORBERG: Okay. Can I leave that aside for a minute? That may just provide some sort of different coloring on what I was trying to say, as opposed to change it markedly. And that is, the idea of the quest and the *goal* of the quest essentially remains the same over the last 30 or 40 years. I mean, I don't want to talk about degrees, I just want to talk about the people in AI.

WINOGRAD: For some subset of people that remains the object.

NORBERG: People who commonly called themselves AI people.

WINOGRAD: Not *all* of those people.

NORBERG: That's where I'm trying to make sure I understand it, so as to not misrepresent the situation. But if you'll

allow me, for a moment, to state that the quest is essentially the same. What has happened over the last 30 or 40 years, it seems to me, is that the methods have both broadened in scope and deepened in sophistication, and that what we have learned in AI as a result of that broadening and deepening is that we make connections now with fields that we never thought we would make connections with before. And as a result, somebody in pattern recognition in an electrical engineering department can now consider themselves to be in AI, because they happen to use some of the methods of AI - or maybe not, as the case may be - but are not in the AI area working on specific methods for the deepening and broadening, further deepening and broadening, of those methods. Does that make any sense, or did I sort of come around in a circle?

WINOGRAD: Let me re-phrase it my way and see if you think it's the same thing. Let me drop the question of who's "in AI" - that is multi-dimensional, it's a complicated thing. What's happened is that the sort of view which is exemplified by the MIT AI lab, which was, "Nothing that was done before is going to be very relevant. We've got this new way of doing things. Everything has to be invented from scratch, and if you're smart you could do it. Background isn't relevant, it's pure cleverness that counts. And, if we do enough of that, we're going to have full intelligence." That is what I say the view was. People have realized now two things and this may be what you're calling broadening and deepening, whatever. One is, in fact, pure cleverness isn't enough.

NORBERG: That's right.

WINOGRAD: And understanding the history of what's gone on in logic, what's gone on in statistical processing and patterning - we're going to need all that. The second is recognizing that what we're going to get out of it may not be "full intelligence," but rather specific capabilities. So, it may be that applying the techniques that we develop, let's not worry about whether it's like human vision really, or whether it really gives full problem solving, but it's a great way to do this task, or to put in the factory, or whatever. And a lot of people have taken that - you might say those people, by the quest definition, have dropped out of AI. They're now doing "knowledge engineering." Here's a magazine I'm on the board of called *AI expert*. These people don't have the quest. These people are building stuff for industry.

NORBERG: Exactly.

WINOGRAD: They're using all of the apparatus that AI people built. But in the end what is the sort of goal here is to say, well, we can put this into a factory, we can put this into a whatever.

NORBERG: But this is true of other disciplines in their growth. If we think about the physicist, for example, and go back to your Kuhn example, this is normal science now, in Kuhn's definition - the development of expert systems that can go here and there. And the people who are still after the quest are the people in theoretical physics, perhaps . . . a unified gravitational principle which will apply to both relativistic and non-relativistic standards, and so on. They maintain this sort of goal of trying to understand the entire universe, whereas other physicists are just trying to make particles go faster. But they're all physicists. So, is it possible that the AI people, that we really want to call AI people, are the ones who still subscribe to the quest, and that everybody else is doing normal science and they're computer scientists today.

WINOGRAD: Well, I guess AI is . . .

NORBERG: Which is, after all, the umbrella which we all sort of work under.

WINOGRAD: There's two different issues. One is normal science versus . . .

NORBERG: Revolutionary science.

WINOGRAD: Revolutionary science. One is trying to find the basic principle versus applying things to make them work. Typically, you distinguish physics and engineering.

NORBERG: That's true.

WINOGRAD: Physicists, even the normal science physicists are trying to get at the basic principle. The engineers are the ones that are building bridges that don't fall down, right? Now, what do you call somebody who is an engineer applying the principles of AI? Are they "in AI" or not. My view is that most of what's called "computer science" is really a branch of engineering. In this university it is; we're in an engineering school.

NORBERG: Yes, I'm struggling with that myself at the moment, trying to decide what in the hell computer science really is.

WINOGRAD: You're not the only one. [laughs]

NORBERG: Well, I just want to make a succinct statement to guide the reader.

WINOGRAD: In the end, of course, it's not interesting to say what's the *real* answer to who's in AI, because that's going to be defined by a community who chooses to hold that word against some other community who wants to grab it, or to keep their identity. It's a label which provides an identity, and the question is which community for which identity wants the label? And I don't know whether Rumelhart wants to be called AI or not. Of course, there are obvious factors, like you don't want to be called the name of the program which has been canceled for funding. You want to be called the name of the program which has just been started.

NORBERG: Indeed.

WINOGRAD: So, he may say, "Oh, no, I'm not doing AI." Right? AI isn't where it's at anymore. "Don't fund those guys who call themselves AI." And in the end that's what determines how the word gets used. It gets used because people put it in proposals and put it on the title of their books, and so on, which is because it serves all these different purposes to identify with certain other people and certain other projects, and certain other things.

NORBERG: Okay, but now you're defining it in a sociological sense.

WINOGRAD: See, my view is all language ultimately is used in a sociological sense.

NORBERG: Well, if that . . .

WINOGRAD: That's my view of language.

NORBERG: If that's the case we'd have the problem in every area and not just with AI and computer science.

WINOGRAD: What's physics? I think you could find people who argue about whether what they're doing really is physics or really isn't physics.

NORBERG: It's rare. It's rare to find that.

WINOGRAD: It's rare because it's, of course, much older.

NORBERG: Yes.

WINOGRAD: Those arguments have been . . .

NORBERG: Well, we assume they are over, but I think what we'd find is that people argue for their research specialty, and therefore they are high energy physicists, or whatever. And it would seem to me that that is where we're moving in computer science with AI. AI is becoming a subset, if you like, or sub-field of computer science, like database management, like architecture, and so on. Whereas, earlier, with the quest before them as the constant goal to be sought, computer science was irrelevant. It was just a tool they could use. Is that fair?

WINOGRAD: That was the view.

NORBERG: A common view between us, that we understand the history that way.

WINOGRAD: Whether computer science was irrelevant depends upon what you call computer science. I would say AI was *always* doing computer science, it's just a realization, not a shift in terms of what was getting done. I mean, symbol processing *is* computer science.

NORBERG: Let me ask you two last questions, both of them are quick. I noticed in some of your recent publications that you're affiliated with the Center for the Study of Language and Information here. What is that? How did it come about?

WINOGRAD: Somebody's actually written a whole dissertation, a two-volume dissertation on that question.

NORBERG: And available here at Stanford?

WINOGRAD: No, this is - do I have it? I may have a copy of it. A guy in Holland, or Denmark, I've forgotten which, who came over here and studied it. I'll give you the quick version; if you're interested I can track down that dissertation. The history goes back to the Rand Corporation, which at one point got too big for some limit the government set on think tanks and spun off something called the System Development Corporation.

NORBERG: Oh, this is the thing they funded. Okay, I know about that.

WINOGRAD: Well, the answer is CSLI is Charlie Smith's baby.

NORBERG: Okay. My last question. The implication in a number of things that you said in the course of the last two hours is that you really had very little contact with the DARPA people, the funders, and so on. Is that fair?

WINOGRAD: That's fair, and it's for two reasons, at least. One is, in that era, and I'm talking up until - well, certainly when I was at MIT and when I came here to Stanford, - the dealing with DARPA - well, it was ARPA - it took me a long time to get into the habit of switching it the other way - was all done by the higher-level people. All you knew is that, you know, Minsky came back every year with a pot of money. And you didn't need to worry about it. They didn't come for demos. You didn't have to write papers for them. Minsky said, "They're paying us. Go ahead and do what you want." So, there was no necessity from the point of view of an individual researcher to even pay attention to that; it was just taken care of. The second was, when I came out here, I got involved at Xerox-PARC, so my research was really not being funded out of Stanford at all, so my, you know - I could tell you a lot about the internal research funding at Xerox; it's a different world. The third is political. I was a child of the '60s, right, and always had a lot of questions about what happens when you let military funding be the driving force in a discipline, not even the direct influence. Clearly some of the things that DARPA has funded have been great. The networking has been very useful. AI I might argue on intellectual grounds as to the direction it took, but I would have had the same arguments if NSF had done it.

NORBERG: Sure.

WINOGRAD: But I think that, in a larger picture of how the country and Congress views research - and we're going through this struggle right now. But there are long-term problems with having the military involved, and I've always had a very queasy relationship, I guess, to military funding. In fact, I haven't had DARPA funding. I haven't had any military funding, since well, basically, when I came out here I had it in this sort of umbrella. I never wrote a proposal to DARPA myself. When I came here I was funded under that umbrella by John McCarthy for a year or two, and that was it. I got some funding from System Development Foundation, I've had some funding from NSF, and I, basically, didn't have much research funding because I was doing research over at PARC, and because I was writing books on philosophy, where you don't need a lot of money, and so on. I'm just now trying to find ways to crank up some money that isn't military money for some of these CSLI things. Now of course, the university is swinging from government to industry funding anyway, so going out for industry funding is the thing they're really interested in, as

opposed to getting more military contracts. So, partly because of the environment and partly because of my own hesitations, I just stayed away from it.

NORBERG: So you had very little, if any, interaction with any of those people. Bob Kahn, Saul Amarel, other than Saul's technical work on representation?

WINOGRAD: I had no interaction with them at all *qua* DARPA. I talked to Saul at a presentation occasionally. I don't think I met Bob Kahn until after he'd left DARPA.

NORBERG: Well, I assumed that from reading your writings and seeing the acknowledgements, and so on, and therefore I had a view that we really wouldn't have to consider that.

END OF INTERVIEW