HUMAN-COMPUTER INTERACTION, 1987-1988, Volume 3, pp. 3-30

A Language/Action Perspective on the Design of Cooperative Work

Terry Winograd
Stanford University

ABSTRACT

In creating computer-based systems, we work within a perspective that shapes the design questions that will be asked and the kinds of solutions that are sought. This article introduces a perspective based on language as action, and explores its consequences for system design. We describe a communication tool called The Coordinator, which was designed from a language/action perspective; and we suggest how further aspects of coordinated work might be addressed in a similar style. The language/action perspective is illustrated with an example based on studies of nursing work in a hospital ward and contrasted to other currently prominent perspectives.

An earlier version of this article was presented at the Conference on Computer-Supported Cooperative Work, Austin, TX, December 1986.
Author's present address: Terry Winograd, Computer Science Department, Stanford University, Stanford, CA 94305.

CONTENTS

1. PERSPECTIVE AND DESIGN
2. THE LANGUAGE/ACTION PERSPECTIVE
3. THE PRAGMATICS OF LANGUAGE ACTION
   3.1. Speech Act Theory
   3.2. Conversations for Action
4. DESIGNING CONVERSATIONS FOR ACTION
   4.1. Tools for Conversing
   4.2. Retrieval and Monitoring
5. CONVERSATIONS IN A WORK SETTING
   5.1. Conversations for Action
   5.2. Conversations for Clarification
   5.3. Conversations for Possibilities
   5.4. Conversations for Orientation
   5.5. The Larger Web of Conversations
6. SEMANTICS
   6.1. What Domains of Distinctions Are Taken as Background?
   6.2. How Do New Distinctions Emerge?
   6.3. How Are Distinctions Indicated?
7. BLINDNESSES OF THE LANGUAGE/ACTION PERSPECTIVE
8. CONCLUSION

1. PERSPECTIVE AND DESIGN

Within the community concerned with the design of computer systems, there is a growing recognition of the importance of the designer's perspective—the concerns and interpretations that shape the design, whether they are articulated explicitly or are just part of the unexamined background of the work. A perspective does not determine answers to design questions but guides design by generating the questions to be considered.

Many writers have identified useful perspectives on computer-based systems, and several classification schemes have been proposed (see, e.g., Kling, 1980; Malone, 1985; Nygaard & Sørgaard, 1985). As we will discuss in section 7, no one perspective covers all of the relevant concerns. System design is a complex web. It can be picked up from any one point, and the others will follow along. If we start from implementation considerations (e.g., available hardware functions), we will eventually have to define user interfaces. If we start by considering user interaction, we must eventually build concrete implementations that can run effectively on the hardware. But although the full range of perspectives must eventually be considered, the outcome will differ depending on where we start.

This article presents a particular perspective; one that takes language as the primary dimension of human cooperative activity. It draws on work devel-

oped by Flores and his colleagues at Stanford University, Logonet, and Action Technologies (Flores, 1981; Flores & Ludlow, 1981; Winograd & Flores, 1986), and has been the basis for designing commercially successful computer systems. By starting with a language/action perspective, we have found it possible to create systems that can be effective in getting work done, whenever that work involves communication and coordinated action among a group of people.

We illustrate the language/action perspective through an example: the nursing work in a hospital ward, as studied by Kaasbøll (1986, 1987). This example was chosen for its careful description and analysis of the setting and the structure of work. No computer-based systems have been introduced into the work or designed in detail. The goal here is to illustrate the questions and concerns that would guide the design of such systems.

2. THE LANGUAGE/ACTION PERSPECTIVE

One useful way to identify a perspective is by its declaration of what people do. From a language/action perspective we say that *people act through language.* As a contrast, consider the more predominant perspective that *people process information and make decisions.* Of course, everyone in an organization can be described as doing both, but there is a difference of focus.

Consider a situation in which a hospital nurse calls the pharmacy, finds out what drugs are available, and orders one of them for a patient. From an information-processing perspective we could focus on the database of information about the drugs and on the rules for deciding which drug to order. From a language/action perspective, we focus on the act of ordering and on the patterns of interaction in related conversations, such as the preliminary conversation about drug availability and the subsequent conversation that unfolds in the process of fulfilling the order. From other perspectives we might consider such things as the personal relationship between nurse and pharmacist, the cost-effectiveness of making the communication over a phone, or the legal status of orders placed by a nurse.

For a perspective to have analytical value, its focus on particular concerns must be combined with a systematic conceptual framework and methodology. For issues of cost-effectiveness, we would turn to economic theory. For information processing, we would look to theories of information and decision making. The language/action perspective rests on theories of language, but not linguistics in the rather specialized sense that is often understood. We are not primarily concerned with the details of natural language utterances, but with the issues of form, meaning, and use that are common to all human communication. We use the word *language* rather than *communication* to emphasize the relevance of symbols and interpretation, and also to avoid the connotations of communication theory, which has come to stand for a rather specialized mathematical approach.

As the following sections demonstrate, there is a broad view of language activity, which includes a wide range of interactions with computers. The theories grow out of previous work in linguistics, but go beyond it and are still being developed. Our own work on design has been intermingled with research on linguistic theory (Winograd & Flores, 1986).

As a broad framework for outlining a language/action perspective, we will adopt and extend the traditional subdivisions of linguistic theory: *syntax, semantics,* and *pragmatics.*

Syntax is the structure of the visible (or audible) forms of language. The syntactic rules (or grammar) of a language determine the basic elements (letters, words, etc.) and the ways in which they can be combined. In an extended sense, one can talk about the syntax of an equation, a spreadsheet, or an invoice, or even of an event, such as buttoning a menu item on a screen. What distinguishes syntax from other levels of analysis is that it does not take into account interpretation or meaning.

Semantics is the systematic relation between structures in a language and a space of potential meanings. It includes the definitions of individual elements (e.g., words) and the meaning that is generated by combining them (e.g., the meaning of "Jill sees Bill" as different from "Bill sees Jill"). In extension, one can talk about the semantics of a blank on a form appearing on a workstation screen or the semantics of an operating system command.

Pragmatics deals with issues of language use. A classical example is "It's cold in here" spoken by a master to a servant. Although the literal meaning is a statement about the temperature, the intent is to evoke an action by the servant. Our primary interest lies in this aspect of language—its role in evoking and interpreting actions.

Modern linguists have tended to adopt a cognitivist approach (Haugeland, 1981), formalizing the structure of an individual language user's knowledge and mental processes. Our perspective leads us to deal with these three aspects of language in the reverse of the standard order—we take issues of meaning to be critically dependent on considerations of language action and context, and syntax to be of interest primarily in its ability to reflect meaningful distinctions in conversation.

3. THE PRAGMATICS OF LANGUAGE ACTION

The language/action perspective emphasizes pragmatics—not the form of language, but what people do with it. The theory of speech acts is a starting point for developing the larger picture of the following sections.

3.1. Speech Act Theory

Austin (1962) noted that not all utterances are statements whose truth or falsity is at stake. Performatives, such as "I pronounce you husband and wife"

are actions, which can be made appropriately or not, but which are neither true nor false in a simple sense. Similarly, the language actions of commands, questions, and apologies are not descriptions of a nonlinguistic world.

Searle (1975) identified five fundamental illocutionary points—things you can do with an utterance:

Assertive	Commit the speaker (in varying degrees) to something's being the case—to the truth of the expressed proposition.
Directive	Attempt (in varying degrees) to get the hearer to do something. These include both questions (which can direct the hearer to make an assertive speech act in response) and commands (which direct the hearer to carry out some linguistic or nonlinguistic act).
Commissive	Commit the speaker (again in varying degrees) to some future course of action.
Declaration	Bring about the correspondence between the propositional content of the speech act and reality (e.g., pronouncing a couple married).
Expressive	Express a psychological state about a situation (e.g., apologizing and praising).

Three points deserve note:

1. The illocutionary point of an utterance is interpreted by speaker and hearer in a background. A commissive need not include the words "I promise" or "I will," but can be "I guess" or "a dollar" (in response to "Can you give me anything?") or just a facial gesture. The identification of a language act depends on the backgrounds of speaker and hearer, and is always open to differences of interpretation. "It's time for lunch" might be an assertive or a directive, depending on who says it to whom in what circumstances.

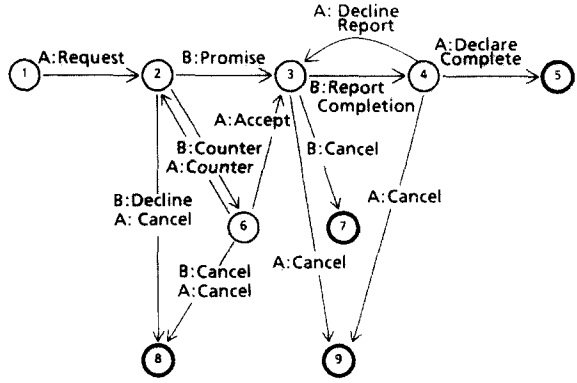
2. Directives and commissives (which will informally be called "requests" and "promises" here) always deal with a future action. They differ in whether the action is to be taken by the speaker or the hearer.

3. Speech acts take effect by virtue of public declaration—by mutual knowledge of hearer and speaker that the act has been made. This is especially obvious in the case of declarations and expressives (e.g., an apology muttered but not heard is not an apology), but is equally true of the others.

3.2. Conversations for Action

Speech acts are not unrelated events but participate in larger conversation structures (Flores, 1981; Flores & Ludlow, 1981). An important example is the simple "conversation for action," in which one party (A) makes a request to another (B). The request is interpreted by each party as having certain

Figure 1. **State transition network representing a conversation for action initiated by a request from speaker A to speaker B. The circles represent conversation states and the labeled lines represent speech acts. Heavy circles represent states of completion. (Adapted from Winograd & Flores, 1986, p. 65.)**



conditions of satisfaction, which characterize a future course of actions by B. After the initial utterance (the request), B can accept (and thereby commit to satisfy the conditions), decline (and thereby end the conversation), or counter-offer with alternative conditions. Each of these in turn has its possible continuations (e.g., after a counter-offer, A can accept, cancel the request, or counter-offer back). The overall structure is diagramed in Figure 1.

This diagram is not a model of the mental state of a speaker or hearer, but shows the conversation as a "dance," in which the conversation steps proceed toward mutual recognition that the requested action has been done or that the conversation is complete without it having been done. The basic logic represented here deals with the central progression of acts. Other possibilities not shown in the diagram can emerge in related conversations, such as those in which the conversational acts themselves are taken as a topic. For example, a speaker might question intelligibility ("What, I didn't hear you") or legitimacy ("You can't order me to do that!").

If B commits to fulfill a request (moving to State 3), the natural continuation is that at some later point B reports to A that the conditions of satisfaction have been met (moving to State 4). If A declares that he or she is satisfied, the conversation reaches a successful completion (State 5). On the other hand, A may not interpret the situation in the same way and may decline the report, declaring that the conditions have not been met and thereby returning the conversation to State 3. In any state, either party may propose changes to the conditions of satisfaction or may back out on the deal, moving to a state of completion (States 7 and 9) that does not include satisfaction of the orginal request.

Several points about the conversation structure deserve note:

1. We use *conversation* in a very general sense to indicate a coordinated sequence of acts that can be interpreted as having linguistic meaning. It need not be a spoken conversation, or even involve the use of ordinary language. A doctor who writes treatment requests on a patient form is engaged in a conversation with the nurse who will administer the treatments, even if they never speak face to face. Certain kinds of requests are made implicitly on the basis of a long-term declaration. A manager does not explicitly request each worker to come to work each morning, although the conversation proceeds (in those cases where there is a breakdown) as though he or she had. The recurrent request is listened to as an effect of the declaration "You're hired" within a shared understanding of common practices.

2. The conversation is initiated by a request (there is a similar network for conversations initiated by an offer), and therefore is rooted in the anticipation of some future action.

3. At each point in the conversation, there is a small set of possible action types, determined by the previous history. Each type has unlimited possibilities for detailed content. For example, a counter-offer action specifies particular conditions of satisfaction.

4. All of the acts are linguistic—they represent utterances by parties to the conversation (or silences that are listened to as standing for an act). For example, the act that normally follows a promise is a report of completion (an assertive speech act) from the promisor to the requestor. It is followed by a declaration by the requestor that the action is satisfactory (or that it is not). The actual doing of whatever is needed to meet the conditions of satisfaction lies outside of the conversation structure.

5. Many acts are "listened to" without being explicit. If the requestor can recognize satisfaction of the request directly, there may be no explicit report of completion. Other acts, such as declaring satisfaction, may be taken for granted if some amount of time goes by without a communication to the contrary. What is not said is listened to as much as what is said.

6. Conditions of satisfaction are not objective realities, independent of interpretations. They exist in the listening, and there is always the potential for differences among the parties. This can lead to breakdowns and to subsequent conversation about the understanding of the conditions.

7. There are states of completion (the heavy circles in Figure 1) in which it is mutually recognized that neither party is waiting for further action by the other. All other states represent an incomplete conversation. Completion does not guarantee satisfaction. For example, if the promisor cancels after the promise is made, the conversation is completed without the original request being satisfied.

8. The network does not say what people should do, or deal with consequences of their acts (such as backing out of a commitment). These are important human phenomena but are not generated in the domain formalized in this network.

Conversations for action are the central coordinating structure for human organizations. We work together by making commitments so that we can successfully anticipate the actions of others and coordinate them with our own. The emphasis here is on language as an activity, not as the transmission of information or as the expression of thought. Although people think when they use language, and they often describe their world in language, the relevant structures for analysis here are the language acts and the conversations into which they are woven. In applying this to computer system design, we are not concerned with duplicating the knowledge or thought patterns of people, but with the structure of their interactions and the embedding of those interactions in computer systems.

4. DESIGNING CONVERSATIONS FOR ACTION

We will illustrate the relevance of this analysis to computer systems by describing The Coordinator,[1] a first-generation conversational system currently used for everyday communications in sales, finance, general management, operations, and planning functions in organizations of a variety of sizes and types. This system provides facilities for generating, transmitting, storing, retrieving, and displaying messages that are records of moves in conversations. However, unlike electronic mail systems that take "messages" and "information" as their starting points, it is based on the conversation theory outlined earlier.

4.1. Tools for Conversing

The user interface of The Coordinator is menu driven. The primary menu for conversing is shown in Figure 2.

Some of the menu items indicate new actions the user may take. Others bring up displays of the records of conversations maintained by the system. Let us look first at ways of opening a conversation for action (answering is

[1] The Coordinator is a workgroup productivity system created by Action Technologies, Inc., available for IBM PC-XT/AT™-compatible machines. The description here focuses on the conversation manager, which is one part of an integrated system that also includes word processing, formatting, calendar maintenance, and communication over modems and local area networks (LANs). The Coordinator is a registered trademark of Action Technologies. The interface design is copyrighted, and aspects of it are reproduced here by permission. A patent is pending on the system's conversation manager.

Figure 2. **Converse menu from The Coordinator. (Reprinted by permission from Flores, Bell, Graves, & Ludlow, 1987.)**

CONVERSE	
OPEN CONVERSATION FOR ACTION	REVIEW/HANDLE
Request	Read new mail
Offer	Missing my response
	Missing other's response
OPEN CONVERSATION FOR POSSIBILITIES	
Declare an opening	My promises/offers
	My requests
ANSWER	Commitments due: 24-Sep-87
NOTES	Conversation records

discussed in this section, and conversations for possibilities are discussed in section 5).

Rather than providing a uniform command to initiate a new message, The Coordinator system provides options for opening conversations that have different implicit structures of action. When "Request" is selected, templates appear prompting the user to specify an addressee; others who will receive copies; a "domain," which groups or categorizes related conversations; and an "action" description, corresponding to the subject header in traditional mail systems. The text of the message is prompted with the phrase "What is your request?," to which the user can enter any text whatsoever. The system makes no attempt to interpret this text, relying on the users' understanding and cooperation that the message is properly identified as a request. This is a key design issue: Let people do the interpretation of natural language, and let the program deal with explicit declarations of structure (such as a user's declaration that this is a request). This leaves users free to communicate in ordinary language that depends on the background of the reader. A perfectly understandable request might contain the single word "Noon?" if the participants have a shared understanding (e.g., they often go to lunch together).

When the user signals that the text is complete, the system prompts for three dates associated with the completion of the action: a respond-by date, a complete-by date, and an alert date. Date entries are optional, but experienced users almost always include one or more of them. Not only do specific dates provide a structure for retrieval and for monitoring completion, but their use plays a surprisingly large role in producing effective conversations. Although we will not emphasize this aspect in this article, the design of The Coordinator system grew out of Flores's work in training people in "communicative competence" (Flores & Graves, 1986a, 1986b). In that work, Flores demonstrated that people's ability to communicate effectively (with or without support from computer systems) is improved when they develop facility in distinguishing the kinds of commitments people make in conversations for

Figure 3. **Menu for responding to a request. (Reprinted by permission from Flores, Bell, Graves, & Ludlow, 1987.)**

SPEAKING IN A CONVERSATION FOR ACTION	
Acknowledge	Promise
Free-form	Counter-offer
Commit-to-commit	Decline
Interim-report	Report-completion

action and the dimensions of time associated with the completion of those conversations.

When a user of the system receives a request (the details of message transmission and retrieval will not be discussed here), he or she has the option of responding by selecting "Answer" from a menu. This pops up a subsidiary menu as shown in Figure 3.

This menu is automatically generated by a conversational state interpreter from a network like that of Figure 1. The first three items in the right-hand column ("Promise," "Counter-offer," and "Decline") represent the actions available to the responder (B) in State 2. The fourth choice ("Report-completion") is an action available in State 3, after B has promised. In some cases, it will turn out that B has already done what A requested, before having responded to the initial request. In that case, the "Promise" act is implicit, and "Report-completion" is the next overt communication.

The left-hand column introduces conversation acts concerned with the conduct of the conversation itself, which do not advance its state. "Acknowledge" lets the requestor know that the request was received. "Free-form" allows any kind of communication relevant to the conversation that does not fit into the formal structure—most frequently, notes, comments, and questions. "Commit-to-commit" would be conveyed in natural language with sentences such as "I'll let you know by Thursday if I can do it." That is, the speaker is committing to take the next conversational step (promising or declining) by a specific time.

When any answering action is selected, a new message is automatically generated with markers corresponding to the choice of act and with a generic text. For example, if the response is "Promise," the initial message is "I promise to do as you request," but for "Counter-offer" is "No; I counter-offer." The user can augment or replace this text using embedded word-processing facilities. Experience has shown that a surprising number of messages need only the initial pro forma composition. The message initiating a request or offer needs to contain text that describes the action, such as "Can you send me that report we were talking about?," but often the subsequent steps can be made by simply selecting the appropriate menu item and hitting the button that sends a message.

Figure 4. **Answer menu generated in continuing a promise. (Reprinted by permission from Flores et al., 1987.)**

SPEAKING IN A CONVERSATION FOR ACTION	
Free-form	Cancel/New-promise
Interim-report	Cancel
	Report-completion

Whenever "Answer" is selected, the menu displays only those actions that could sensibly be taken next by the current speaker. State 2 of Figure 1 shows a "Cancel" action by A, in which the request is withdrawn. This will appear on A's menu, but not on B's. Or, for example, after making a promise in a conversation, the next time B selects "Answer" in that conversation (assuming no intervening action by A), the menu offered will be as shown in Figure 4.

At this point, B no longer has the option to decline (having already promised), but can "Report-completion" (moving to State 4) or "Cancel" (moving to State 7) with or without initiating a new promise.

The Coordinator has no magic to coerce people to come through with what they promise, but it provides a straightforward structure in which they can review the status of their commitments, alter those commitments they are no longer in condition to fulfill, make new commitments to take care of breakdowns and opportunities appearing in their conversations, and generally be clear (with themselves and others) about the state of their work.

4.2. Retrieval and Monitoring

The structure and status of conversations is the primary basis for organizing retrieval and review in the system. To put it simply, the structure is organized to provide straightforward and relevant answers to the implicit question "What do I have to do now?"

In the main menu of Figure 2, under the heading "REVIEW/HANDLE" we find items such as "Missing my response," "Missing other's response," "My promises/offers," and "My requests." When one of these is selected, the user is presented with a listing of conversations matching the selected item. Several things are of note:

1. The basic unit of work in the system is a conversation, not a message. In conventional electronic mail systems, messages in a conversation are often linked by conventions such as the use of "Re:" in headers. For The Coordinator, each message (including a "Free-form") belongs to a particular conversation. The retrieval structure is two-level, with the user first identifying a conversation, then selecting particular messages within it to be displayed.

2. The explicit use of conversation theory in the generation of messages makes it possible for retrieval to be based on status. There is a menu selection that selects and displays conversations in response to the question "In which conversations is someone waiting for me to do something?" or "In which conversations have I promised to do things?" Note that these are different. For example, if you make an offer to me, then our conversation is in a state in which the next move characteristically belongs to me, but I have made no promise to you.

3. The distinction between "open" and "closed" conversations is used to filter out those to be retrieved. Unless the user designates otherwise, The Coordinator will display only those conversations that are still open to further action (i.e., not in one of the final states as shown by heavy circles in Figure 1).

4. Explicit completion and alert dates are used for time-oriented retrieval. The item "Commitments due" on the menu allows retrieval of all conversations that need some action (either a response or a completion) on a date entered by the user. There is an additional menu that allows retrieval on precise combinations of dates, domains, and people involved in different conversational roles (e.g., the things I have promised to get done next week regarding programming). The calendar subsystem is integrated so that all of these items can optionally appear at the appropriate places in a personal calendar, along with more conventional entries such as meetings and appointments.

The Coordinator is an example of basing a system on theories of language without attempting to program "understanding." All of the interpretations (e.g., that a particular message is a request, or that it should be done by a certain time) are made by the people who use the system, guided by appropriate menus and prompts. This is not experienced by users as an extra job of annotating, but in fact replaces typing parts of the contents with more direct and structured interactions, which are often more efficient. It is a generic tool in the sense that a word processor is—intended for a particular kind of communication, without regard to topic. A word processor is not equally well suited to generating all kinds of character sequences, but is specially designed for the words, sentences, paragraphs, and the like of ordinary written text. Similarly, The Coordinator system is not built for arbitrary sequences of messages, but for the requests, promises, and completions that are at the heart of coordinated work.

5. CONVERSATIONS IN A WORK SETTING

Conversations for action (CfA) form the central fabric of cooperative work. However, many kinds of language acts do not participate directly in the completion of a CfA. Remarks such as "They're planning to remodel the west

wing next summer" need not relate directly to any specific future actions of speaker or hearer. From a cognitive perspective, one might choose to characterize these as "conveying information" without a particular motivation in action. From the perspective of language as action, the primary concern is with the role that all conversations (and all utterances within conversations) play with respect to action and potentials for action. We distinguish several additional kinds of conversation that go along with conversations for action: *conversation for clarification, conversation for possibilities,* and *conversation for orientation.* There is no sharp line between them, but they are accompanied by different moods.

In a conversation for clarification, the participants cope with or anticipate breakdowns concerning interpretations of the conditions of satisfaction for a CfA. The conditions are always interpreted with respect to an implicit shared background, but the sharing is partial and needs to be negotiated. As a simple example, the request "Give the patient some diazine" might evoke responses such as "Right now, or with the morning meds?" or "What dosage?" One can never guarantee that everything is totally precise. Precision is relative to each party's implicit anticipation that the other party will have a sufficiently shared background to carry out the action in a satisfactory way.

In a conversation for possibilities, the mood is one of speculation, anticipating the subsequent generation of CfAs. Specific conditions of satisfaction will emerge in the course of the conversation, and associated conversations for action will be initiated. Many gatherings that are called *meetings* are best conducted in this mood. The meeting is a failure if some action does not come out of the discussion. Some conversations for possibilities are highly routinized. For example, work rounds on a hospital ward is a routine conversation for possibilities, during which the medical team visits each patient and specific requests and commitments are generated.

In a conversation for orientation, the mood is one of creating a shared background as a basis for future interpretation of conversations. This shared background includes specific knowledge, interpersonal relations, and general attitudes. The most obvious examples are meetings labeled *orientation,* in which newcomers begin to develop the understanding that is required to function in the organization. Conversations for orientation are prominent in less formal settings ("shooting the bull"). Although the mood here is not directed towards action, it is important to recognize the importance of developing mutual orientation as the basis for future effective action and for appropriately shared interpretation of language acts.

Each of these types of conversation has its own regularities of structure, which in turn can be reflected in the design of the tools for conducting it. Just as the CfA structure of The Coordinator grew out of experience with conventional message systems, we can apply conversational analysis to the reinterpretation and redesign of other existing systems, such as help systems

(which carry out a limited kind of conversation for clarification); group facilitation systems, such as Colab (Stefik et al., 1987) which are used in generating possibilities; and bulletin board and computer forum systems, which among other things, facilitate conversations for orientation. We will not analyze these in detail, but will use the nursing example to show how conversations appear in the nursing setting and to discuss some of the design considerations.

In the discussion we will comment on details of work on the hospital ward, as outlined in Figure 5. Although no computer applications were developed in that setting, one can imagine an integrated medication information system through which many of the activities would be replaced by actions on terminals (or workstations) at various sites, including the ward, the examining rooms, and the pharmacy. Records needed in places where direct computer access was infeasible could be printed out and posted. The information flow could be redesigned, eliminating redundancies and the need for manual copying or posting of information. It is far beyond the scope of this article to develop a comprehensive design for such a system, but the setting can serve to illustrate our perspective.

5.1. Conversations for Action

In the hospital, there are many different conversations for action, with a variety of visible forms. Some are highly routinized, such as the primary CfA dealing with the administration of medications. Requests are made by doctors (either as standing cures or on the patient-carried paper scraps) to the treating nurse. Reports of completion are represented on the curve sheet, and the declaration of completion is implicit in the doctor's review of the records on his or her next visit. As a precondition for satisfying these requests, the nurse must receive the medicine, and there are CfAs (with the pharmacy) to get the medications, using prescription forms to make requests. In general, conditions of satisfaction are determined in a rigid way by the codes and blanks, perhaps with extra notations in natural language. Acceptance of an offer or request is assumed whenever it is not explicitly rejected. Completion is reported on a standard form that, like all of the other forms, is associated with standards for interpretation, which are learned as part of the relevant professional training. In addition to these routinized CfAs there are unscheduled verbal conversations. For example, a request may be made by a doctor to a nurse at the bedside, with immediate explicit accept, decline, or counter-offer. Completion may be reported later via a note in the patient's chart.

In a hospital, completion of conversations can be a life-or-death matter. There is a highly regularized structure of checks and cross-checks to ensure it, as illustrated in Figure 5. The regularization is both in the form of special

Figure 5. **Activities on a hospital ward. (Adapted from Kaasøll, 1986, p. 3.)**

In a case study described by Kaasbøll (1986), researchers in the Florence project of the Scandinavian research program on System Development and Profession Oriented Languages (SYDPOL) analyzed work on a ward of a Norwegian hospital, from what they call a "systems perspective." They focused on the tasks associated with giving medications in a ward for children with respiratory problems. This figure paraphrases Kaasbøll's verbal description of some of the activities that were analyzed.

Each nurse has a special responsibility as the *team nurse* for a small group of patients.

On the day shift, one nurse (called the *treating nurse*) has the task of giving medicines. Her working day may be characterized roughly by the following sequence: attend the report meeting; give medicines; record the medicines given; take care of children in kindergarten or the dining hall.

The report meeting takes place from 7:45 to 8:00 a.m. One nurse informs the other staff of the status, changes, and activities of each patient during the previous day and night. She has heard a vocal report from the night shift, and she reads the Kardex while reporting. The Kardex contains diagnosis, planning, and evaluation for each patient, and a form with fixed *main patient information.* It is supposed to be up to date. Other staff take notes on their *program sheets.*

Medicines are prescribed by the doctors on *prescription forms* in cures lasting several days or in daily doses. Cures are recorded on the main patient information form and on the *medicine card.*

Medicines are given in a treatment room between 8:00 and 9:30 a.m. Patients enter having been examined by a doctor, carrying with them a scrap of paper on which the doctor has written today's dose and possible changes in cures. Prescriptions for medicines during the intervals between the regular medicine hours are noted on a *premedlist* hanging on the wall in the treatment room.

Some simple lung function tests are also performed in the treatment room to monitor the effects of the medicines. Tests to be taken are written on the prescription form and on a scrap taped to the medicine card or on the program sheet if there are changes. The test results are recorded on special forms.

After having given medicines, the treating nurse brings her papers to the ward office. Together with each of the team nurses, one at a time, she examines the papers. All medicines given are now registered on the *curve sheet.* Changes in cures are recorded in the Kardex, and on the medicine card and eventually on the premedlist. In addition, all sheets are compared for the sake of control.

The patients are processed one by one. The team nurse reads from her papers which medicines are to be given. The treating nurse answers by stating which are actually given, and the state of the patient. One day, when the load on the nurses was relatively low, this activity lasted from 9:30 a.m. to 13:25 p.m. During these 4 hr, at most 30 min were "effective paper work." The rest of the time consisted of delays, either because some of the papers were used by others, or because one of the nurses was engaged in handling interruptions. Only a minor part of the 30 min was used for updating and comparing. The rest of the time was spent on small conversations, initiated by findings in the information nurses were handling. These included:

reporting to each other about the patients' state and activities
deciding what were facts when inconsistencies were found
deciding changes in some medicines after small negotiations
reminding the treating nurse of a test that had been forgotten
investigating why a medicine was not delivered from the chemist's
finding out why a patient had to take a specific test.

activities (the various check lists) and the strict temporal routine. The fact that a particular action will be done at a particular time can be taken for granted on the basis of the daily schedule. The dependence on rigid forms and routines can be viewed as an attempt to assure that conversations proceed smoothly in cases where personal contact is not sufficient. This could potentially be reduced, adding work flexibility, through the use of a conversation-based system in which the monitoring of completion (and coaching towards completing conversations) is incorporated in a communication medium.

There is also the potential to replace routine CfAs with declarations of recurrent responses. For example, rather than responding to each drug request, a pharmacist might establish an automated prescription filling system, which takes the data from the request and activates a mechanized dispensary. This is a common kind of computerization: Computers take over those functions for which precise repetitive rules can be established. In designing these automated systems from a language/action perspective, we are led to consider the potential for secondary conversations. Who declares the distinctions that are embodied in the forms and rules? If the medication request does not match the standard form for designating medications, then who is involved in the conversation for clarification, and how? In conventional system design, there will always be de facto answers to such questions (especially after experience has pointed out the places for breakdown). Through a conversational analysis we can anticipate and design for them.

5.2. Conversations for Clarification

Conversations for clarification are much less regular (as we would expect) and are often verbal. The cross-checking of the various forms also triggers these conversations when the different forms are not directly contradictory, but are open to conflicting interpretations. In designing tools for conversations for clarification, it is important to recognize their relative lack of recurrence. Recurrent differences of interpretation will lead to the declaration of new distinctions or new forms for making requests and commitments that are clear. But, there will always be irregular, unexpected cases, and computer-based systems that provide only rigid forms may make it difficult or impossible to deal with them.

5.3. Conversations for Possibilities

Much of what appears to be useless copying or verification of redundant information on the hospital ward is really a routine way of generating conversations for possibilities. For example, in the review of medications (Figure 5):

> Only a minor part of the 30 min was used for updating and comparing. The rest of the time was spent on small conversations, initiated by findings in the information they were handling. These included: reporting to each other about the patients' state and activities; deciding what were facts when inconsistencies were found; deciding changes in some medicines after small negotiations; reminding the treating nurse of a test that had been forgotten; investigating why a medicine was not delivered from the chemist's; finding out why a patient had to take a specific test.

If the medication review were replaced by an automated process, the opportunity for this kind of conversation could be lost. It could be reinstituted in other ways if it were recognized in the analysis. Similarly, if the curve sheet is replaced with a table of drugs, dosages, and times, it is no longer possible for the nurses to use it as a vehicle of communication (in the notes) for the "small conversations" dealing with specific breakdowns and subsidiary CfAs. It might well be possible to replace it with a better vehicle for these conversations, once their importance is recognized.

5.4. Conversations for Orientation

When asked to comment on the systems analysis, the nurses felt that it did not capture the aspect of their activities that dealt with the total picture rather than with the specifics of particular medications and tests. The existing structure provides explicit routines to allow for open-ended conversation, such as the morning report, in which "one nurse is informing the other staff of the status, changes, and performed activities of each patient during the last day and night." Informal exchanges among the nurses include both explicit CfAs and more general orienting discussion about the total picture. One form of orienting conversation is the telling of stories, whether in the direct line of work or around the coffeepot. Orr (1986) described the importance of relating "war stories" as part of technical training. One has only to spend a short time in the company of medical workers to realize how prevalent this activity is. Computer bulletin board systems sometimes play this role within the community of computer researchers. They often contain extended stories, commentaries, and other forms of conversation that serve for mutual orientation. This would not be directly applicable to the hospital setting, where the workers do not spend long hours in front of a computer terminal, but it illustrates a potential for design.

5.5. The Larger Web of Conversations

Finally, the conversational analysis includes not only those immediately visible, but also the larger web of conversations in which they are situated

(Kling & Scacchi, 1982). One obvious example in the hospital is the legal conversation about the quality of care. All written records are potential evidence in a malpractice suit, and the people who create and manipulate them are aware of this possibility. In addition to the legal conversation, there are ongoing conversations about the hiring, evaluation, and dismissal of employees. Kaasbøll (1986) noted: "If the nurses make mistakes, they may be sued by the patients, and they may be punished by the hospital administration. This gives an incentive for not recording mistakes in the Kardex" (p. 11).

Certain conversations will inevitably go on outside of any written (or electronically stored) system. Explicit records will never correspond to an objective reality, but are the result of declarations by individuals, with their own interpretations and purposes. No computer system can change these fundamental facts about how humans function in organizations, but an explicit understanding of the larger network of conversations can help to recognize the roles that language acts play in a variety of conversations and to match expectations to those roles.

6. SEMANTICS

The preceding sections have introduced conversation types in a very general sense, without considering what the conversations are about. Of course, there is also a high degree of recurrence in content, which is apparent in the amount of organizational communication conducted with forms of various kinds (including electronic forms). When a doctor requests medication for a patient, we can identify the generic action as a "Request" in terms of the CfA structure, and we could use a system like The Coordinator to monitor its completion. From a slightly different angle, we can see it as an instance of a "Medication Order" conversation, which is specified by filling in standard blanks, such as the patient's name, and the identity and quantity of the drug. The doctor could take a single action (e.g., a menu selection) to bring up a display with the relevant items indicated and with some of them initially filled in (e.g., the date). Others might be filled automatically (e.g., the standard dosage when the drug name is given), or with machine aid (e.g., providing generic drug names corresponding to brand names).

None of this is new. Business programs along these lines can be found from the airline counter to the grocery store. We can think of these systems as embodying frozen conversation structures. In designing the forms and interactions, programmers embody their understanding of a specialized conversation structure and a set of procedures for completing the conversations. General facilities for specifying office information systems are described by Ellis and Nutt (1980) and in the more recent research exemplified by Malone, Grant, Turbak, Brobst, and Cohen (1987). A flexible specification formalism for forms (or messages) and relationships among their parts makes it easier to design appropriate forms and blanks and to support automation.

From a language/action perspective, two fundamental issues appear. First, there is the role of conversation structure. Another form, such as the "Nurse's Patient Report," may embody a further action in the conversation (e.g., reporting completion). These linkages can be the basis for retrieval and presentation, as well as the structure for the overall system and the procedures that go with it.

Second, semantics is subordinate to language action. Traditionally, semantics has been described as a correspondence between the forms of a language and some kind of "truth conditions" on the world of which it speaks. The analysis concentrates on deriving meaning from the systematic combination of elements. One takes for granted a collection of basic terms—nouns, verbs, adjectives, and the like—referring to identifiable objects, properties, relations, and events in the world. From the perspective of language as action, words cannot be defined in isolation from a particular conversational setting in which they are used. The distinctions that are reflected by the choices among words arise through recurrent patterns of conversation, in which breakdowns of action lead to new distinctions (Winograd, 1985; Winograd & Flores, 1986).

This is equally true in our extended linguistic perspective. A computer-based form has a syntax in which the individual fields and their fillers are the basic units. The interpretation of a field marked "Status" cannot be based on a general definition of what a status is. It will depend on the context and background of the people who enter, interpret, and use the records. The communication will be effective only to the extent that relevant background is shared. Texts on business data processing discuss the importance of data dictionaries, which prescribe the meanings of the individual records and fields in a database. Behind that activity there are questions as to where definitions come from, how they are represented, and how they are understood by the people who use them. These are analogous to problems in natural language.

6.1. What Domains of Distinctions Are Taken As Background?

Everyone in a normal work setting shares a natural language and a lifetime of cultural experience. The everyday use of language takes this for granted, using ordinary vocabulary along with common technical terminology, such as that of the clock and calendar. Other meanings will be specialized to a professional area such as medicine. The boundary between natural and specialized domains is not sharp—many words are used in both informal and semiformalized, or stylized ways. "The patient is in stable condition" has a technical interpretation distinct from the natural one. In some cases, such distinctions may be set down by formal rules; in others, they are learned through practice.

A crucial part of professional training is learning a jargon—the distinctions

and associated terms that provide a basis for inventing and taking relevant actions. Profession-oriented languages (Kaasbøll, 1986) are an attempt to integrate this specialized language structure into the design of computer-based systems. Kaasbøll (1987) pointed out problems, such as locally used distinctions that are not standard to the field and not immediately available to system designers. For example, the nurses in his study referred to a lung-function test apparatus as "the Ohio," which was its brand name, and they had no more general term. In some cases such matters are of critical importance to conditions of satisfaction. In the medical profession two different kinds of terms are used to describe medications: brand names (e.g., Tylenol) and generic names (e.g., acetaminophen). A request made for a brand name may or may not be satisfied by an equivalent generic, depending on a complex interaction of standard practices and local regulations.

Suchman (1987) described the problems that arise from failing to account for differences in semantic intepretations when designing user interfaces. In a user-friendly interface for a copier, language about the machine and the user's actions appeared in various forms on the screen, using distinctions and words that made perfect sense to the copier-designers, but that led to serious breakdowns for users without the same background. In one case, failure to distinguish the "document cover" from the "bound document aid" led to interpreting help instructions in a way completely different from what the designers intended.

The point here is that the system designer cannot assume that the semantics—the mapping from words to distinctions of interest—will either be natural or follow some existing formal specification of the domain. This is especially relevant in a setting like the hospital, where conflicting languages are already in use (the language of doctors and nurses, ordinary language, etc.).

Along with the specific conversations and forms, the system designer participates in designing the professional language of the workplace. A new computer system will alter the language, both in interactions with the system and in the work around it. Andersen and Madsen (1986) pointed out the change in usage of the word *document* when an indexing system was designed for a document collection, then extended to the whole library. All of the indexed items (including books and magazines) came to be called documents, contrary to prior usage. There is both a danger of creating confusion and an opportunity to shape the conversations and the work itself.

6.2. How Do New Distinctions Emerge?

New distinctions are always emerging because of new breakdowns or anticipation of them. A frequent reason for the failure of computer systems is

that they lock in a set of distinctions without provision for evolution. Gradually people find more and more need to work around the system, leading to complexity and chaos.

For example, based on the manual forms used by the dieticians, a patient's diet might be recorded as one of a fixed set of choices, such as no salt or diabetic. Imagine that a new kind of diet is added, such as limiting the cholesterol within the existing diets. The new distinction is not simply one more alternative but a modification of each of the preexisting choices. We might add a collection of new diets such as no-salt low cholesterol or diabetic low cholesterol, but this is a work-around. If a further qualification (high potassium) is added, the system will begin to bog down. What is needed is an interpretation of diet as combining a set of separable dimensions, instead of as a simple choice. The original system designer could not anticipate this need. Gerson and Star (1986) described the problems that arise and the need for what they call "due process" in maintaining shared understanding.

Kaasbøll (1986) gave an example of a technical term whose meaning was subject to argument and the imposition of authority:

> During a doctor's visit, the doctor and two nurses started a conversation of what "P1" means. P-values are measures of obstructions in the lungs.
>
> Doctor: It is the air in the lungs that counts, not the sounds.
>
> Nurse 1: It is obvious that Peter (the chief physician) has a different opinion of P1 than you.
>
> Nurse 2: One has to remember that there are individual discrepancies between the children, such that P1 does not mean the same for one child as for another.
>
> This discussion can be interpreted as a negotiation over the semantics of P1, and thus as a development of the language as the ward. It can also be seen as part of the power struggle between the two professions involved. (p. 10)

The larger pragmatic analysis of conversations and roles includes the conversations in which meaning is negotiated and their reflection in a computer system.

6.3. How Are Distinctions Indicated?

Much of the traditional work on natural language semantics adopts the idealization of a relatively straightforward compositional mapping from forms to meanings. Put simply, each basic term (word) has a meaning, and each phrase or sentence has a meaning made up from the meaning of its parts in a standard way. Some current research goes further, focusing on the effect of context on meaning. It is based on structured analyses of contexts (both the

linguistic context and the situation of the speaker and hearer) and the relation between those structures and the meanings of utterances. There has been some interest in applying the resulting theoretical framework to nonnatural languages, such as programming languages and human-computer interfaces (Center for the Study of Language and Information, 1984). It is beyond the scope of this article to survey the relevant work. It has significant limitations, as discussed by Winograd (1985).

As an example, consider the meaning of filling in a medicine card listing a patient's medications. From a standard semantic view, each blank would be filled with a term that denoted a particular medication, and the card as a whole (analogous to a sentence) would enumerate all the medications to be given to the patient. This is typically the case, but according to Kaasbøll (1987):

> The sheets were filled in properly during the 30 minutes, except for a couple of observed "missing medicines" on the medicine cards. When asked about the "mistakes" the treating nurse replied: "Oh, but we know he (the patient) is going to have the medicines even if it is not written here. It is erased only because he has been under intensive care for some days." (p. 4)

The issue here is not the exact form of the cards. It could just as well have been a computer system with database entries for medications. The situation-dependence of meaning is in the people who enter and access the data. Accuracy is not just a matter of having the computer keep its records straight. In designing and using a system, it is critical to understand the different potentials for interpretation and either cope with them or modify them through training.

The point of all this from the language/action perspective is to treat the generation and interpretation of semantic distinctions as an activity based on conversations that can be designed and facilitated through the computer. One general principle pointed out by Nygaard (1986) is the importance of having distinctions that are open to new interpretation by the workers. In practical terms, this may be as simple as having a "Notes" field in some data record, that allows the worker to enter (and retrieve) ordinary natural language text, as opposed to fixed fields, in which the distinctions are fixed by system convention. It may also lead to new kinds of structured conversations within the work.

7. BLINDNESSES OF THE LANGUAGE/ACTION PERSPECTIVE

Technological impacts cannot be fully understood from any one perspective. Each perspective brings forth some concerns and is blind to others. In

designing a coherent system we are guided by a choice of perspective, but success comes from anticipating breakdowns that only become visible from other perspectives. A number of other perspectives will interact with a design generated from concerns of language action:

Implementation. From an implementation perspective we are concerned with issues of hardware, operating systems, languages, data formats, and the like. The vast bulk of the detailed literature on system design approaches problems from this perspective, as it must for practical reasons.

Web of Computing. As Kling and Scacchi (1982) pointed out, we cannot look at the computer system in isolation. The implementation design is part of a larger web of issues surrounding the computer system itself, such as the design, acquisition, installation, maintenance, and hiring and/or training of people to use a system. These include economic, political, and social considerations and each of these has its own domains of conversation, possibilities, and breakdown. In a way, this perspective is opposite implementation—these are the concerns that ultimately must dominate practical choices, but they do not provide a structured basis for creating a design.

Information Processing. Traditional system perspectives have centered on the kinds of information being entered, stored, and accessed, and on the logical rules relating them. As with implementation, this is obviously the perspective from which many details have to be approached. Our relative lack of attention to those issues here does not mean that they are unimportant. Our argument is that, like implementation, they should be looked at in a subordinate way, guided by considerations of the role they play in the structure of language actions by the people using the system.

Roles, Locations, and Materials. Holt, Ramsey, and Grimes (1983) presented a role/activity theory that focuses on people's roles (which specify sets of behaviors) and on the temporal and spatial structure of their potential interactions. Holt (1986) noted that

> What first stands out in any work environment is its architecture—that is to say its spatial-functional organization. . . . Functional proximity is what relates work places to each other. It is the relation which constrains and organizes the movement of people and materials. . . . (p. 1)

From a language/action perspective we can understand roles in terms of potential for entering into particular recurrent conversations, but we do not have any tools for describing the distribution of materials or the physical

potential for interaction. For example, there may be a critical difference between putting a single terminal at the main nursing station, putting a terminal in each examining and activity room, and having one available by every bedside. In looking at the role of an individual, we need to recognize that he or she can be in only one place at one time, and is limited in the ability to move from place to place. In a way this sounds mundane, but it is all too easy to design a system that would work wonderfully—if the nurse would walk over to the nursing station before giving each patient medication—but that doesn't succeed in practice.

Authority. An important aspect of every human organization is the distribution of authority and the mechanisms by which it is maintained. The introduction of a new technology can perturb this structure in a variety of ways: facilitating detailed monitoring of performance, making it possible for subordinates to work in ways that are not understood by their superiors, and opening possibilities for communication that crosses lines of authority. An analysis of conversational roles can identify particular individuals as having the ability to initiate or respond in certain conversations, and this structure is the practical consequence of authority. But the mechanisms by which authority is established and maintained go beyond this. In contrasting the "tool perspective" to a more traditional systems perspective, Ehn and Kyng (1984) focused on this issue, looking at ways to maintain the autonomy of workers in the face of computer-based changes that can potentially be used to expand centralized authority.

Group Interests. Work is not carried out by a homogeneous collection of individuals. Every work setting contains groups with collective interests, which can be affected by the introduction of computer systems. The redesign of work is a negotiation among the groups already doing and supervising the work, and the results will be shaped by the interests of these groups and the compromises among them. This kind of issue is often critical to system design, for example, between journalists and typographers in newspaper publishing systems (Howard, 1985) and between librarians and clerks in libraries (Andersen & Madsen, 1986). In the hospital, there are powerful constraints on the appropriate role behavior of doctors and nurses. The structure of interactions within the organization maintains this identity, and changes can threaten it. In our example, one might imagine merging the various records and thus eliminating the Kardex. But, according to Kaasbøll (1987):

> Nurses are traditionally a paraprofession subordinate to the physicians and their medical knowledge. . . . In the nurses' struggle for acceptance of nursing as a profession, the theoretical concept "nursing process" and its practical documentation in the Kardex is of central importance for

> developing nursing as a science on its own. In this struggle, the Kardex as a basis for nursing decisions may be seen as the nurses' answer to the physicians' medical records. (p. 35)

Conflict. Most of the organizational models applied to information system design are based on the assumption of shared goals among the participants. In real organizations there are always conflicts among competing goals held by different individuals and groups. In some cases this is institutionalized (as in contractual labor–management relations or internal market competition in a firm), but it is always present. A system that assumes idealized cooperation may easily fail as the result of behavior that the systems analyst might label as stupidity, sabotage, or just plain human stubbornness. An analysis that takes conflicting interests into account is not a vain attempt to dissolve them, but instead is an attempt to channel those interests into explicit forms of mutually agreed-upon negotiation.

The language/action perspective establishes a structure for negotiation, based on a theory of cooperation that assumes the willingness to enter into serious conversation, without assuming shared goals or agreement. A conversation for clarification, for example, might involve each party's negotiating to get a favorable deal, but it can nevertheless result in a mutual agreement. More work needs to be done in intergrating a conflict perspective (Ciborra, 1985; Nygaard, 1986).

Interpersonal Relations. One of the most obvious effects of computer systems is the replacement of face-to-face verbal interaction with computer-mediated exchange. Some of the potential problems can be characterized in conversational terms. A face-to-face interaction that is identified as playing a particular role in CfAs (e.g., medication record entry) often has other components (conversations for possibilities) that are lost when it is replaced with computer interactions. Language acts, in general, can be less effective in the absence of personal relationships. In a study of the introduction of a production planning and control system into a factory, Schneider and Howard (1985) noted:

> In the contributing areas, Production Support personnel are constantly engaged in informal discussions, promises and agreements. . . . Schedulers spend nearly half their time in meetings, competing with their colleagues over shop capacity and priority (one likens the process to "butting heads"). Thus, a major part of the production planning and control process involves the extremely social acts of persuasion, negotiation, and, at times, argument. As one Production Control expeditor puts it, "I'm just one leaf on the tree. I try to go in any and all directions in order to get a part out. It all depends on developing working

> relationships with people in other departments—purchasing, quality control, manufacturing engineering. It's a matter of trust built up over time. Personalities play a big role in it." (pp. 14–15)

In their study, Schneider and Howard showed the pitfalls of trying to redesign the work to eliminate these interactions. Although the language/action perspective focuses on the conversations among individuals, it is structural, not psychological. It asks us to look at the potentials for interaction, but not the motivations and feelings that will lead to what people actually do. Questions of mood, motivation, and personal satisfaction go far beyond anything that has been dealt with here, and are essential to successful design.

8. CONCLUSION

We began with the declaration that a system designer benefits from having an explicit awareness of perspective. A perspective generates concerns and questions, and provides a structured analysis through which they can be addressed. Although every design must eventually confront issues from all perspectives, its overall direction is strongly affected by the ones taken as primary.

We have shown how cooperative work can be interpreted as the generation of language acts and conversations. Experience with The Coordinator has demonstrated the value of this perspective in designing workgroup communication tools. In its capacity as a general medium for conversations for action, it has improved work capacity and effectiveness in a variety of settings. The next step will be to apply the language/action perspective to the design of systems that deal with the recurrent content of conversations, with the other types of conversations, and with the relations linking one conversation to another.

There is little agreement as to what core issues will define the area of research in office systems and computer-supported cooperative work. The fields cannot be defined by particular implementation techniques, or principles of information processing, as these apply to all computer systems. We believe that they are part of a new discipline that focuses on the interaction between the structure of systems and the structure of work, and we anticipate that the language/action perspective will play a major role in its development.

Acknowledgments. My conversations with Fernando Flores have been the basis for my understanding, in the most fundamental ways. Without his teaching, my perspectives would have been far different. I wish to thank the participants in the working meetings on system development at the Center for the Study of Language and Information at Stanford University for their valuable contributions to the evolution of this work. Jens Kaasbøll and Kim Halskov Madsen were especially helpful in giving

me access to and a better understanding of the work being done in the SYDPOL projects. Chauncey Bell, Bradley Hartfield, Françoise Herrmann, Mary Holstege, Judy Olson, and Liam Peyton gave insightful critiques of earlier drafts.

Support. The work presented here was supported by the System Development Foundation under a grant to the Center for the Study of Language and Information at Stanford University.

REFERENCES

Andersen, P. B., & Madsen, K. H. (1986). *How to handle the intangible: Metaphors in design and use of computer systems* (Tech. Rep. No. 87-532). Aarhus University.

Austin, J. (1962). *How to do things with words.* Cambridge, MA: Harvard University Press.

Center for the Study of Language and Information. (1984). *Research program on Situated Language* (Report No. CSLI-1). Palo Alto, CA: Stanford University.

Ciborra, C. U. (1985, August). *Reframing the role of computers in organizations, the transaction costs approach.* Paper presented at the Sixth Annual International Conference on Information Systems, Indianapolis.

Ehn, P., & Kyng, M. (1984). A tool perspective on the design of interactive computer support for skilled workers. In M. Saaksjarvi (Ed.), *Report of the Seventh Scandinavian Research Seminar on Systemeering* (pp. 4-35). Helsinki.

Ellis, C. A., & Nutt, G. J. (1980). Office information systems and computer science. *Computing Surveys, 12,* 27-60.

Flores, C. F. (1981). *Management and communication in the office of the future.* Unpublished doctoral dissertation, University of California at Berkeley.

Flores, C. F., Bell, C., Graves, M., & Ludlow, J. (1987). *The coordinator workgroup productivity system I* [Computer program]. Version 1.5P. Emeryville, CA: Action Technologies.

Flores, C. F., & Graves, M. (1986a). *Domains of permanent human concerns.* Unpublished manuscript.

Flores, C. F., & Graves, M. (1986b). *Designing education.* Unpublished manuscript.

Flores, C. F., & Ludlow, J. (1981). Doing and speaking in the office. In G. Fick & R. Sprague (Eds.), *DSS: Issues and challenges* (pp. 95-118). London: Pergamon.

Gerson, E. M., & Star, S. L. (1986). Analyzing due process in the workplace. *ACM Transactions on Office Information Systems, 4,* 257-270.

Haugeland, J. (1981). The nature and plausibility of cognitivism. In J. Haugeland (Ed.), *Mind design* (pp. 243-281). Cambridge, MA: Bradford/MIT Press.

Holt, A. (1986, August). Primitive man in the electronic work environment. *Conference on electronic work,* Milan.

Holt, A., Ramsey, H. R., & Grimes, J. D. (1983). Coordination system technology as the basis for a programming environment. *Electrical Communication, 57,* 307-314.

Howard, R. (1985). UTOPIA: Where workers craft new technology. *Technology Review, 88,* 43-49.

Kaasbøll, J. (1986). Intentional development of professional language through computerization: A case study and some theoretical considerations. *Proceedings of the IFIP Working Conference on System Design for Human Development and Productivity Through*

Participation (pp. 232-274). Amsterdam: North Holland.

Kaasbøll, J. (1987). *Observation of people working with information: A case study.* Manuscript submitted for publication.

Kling, R. (1980). Social analyses of computing: Theoretical perspectives in recent empirical research. *Computing Surveys, 12,* 61-100.

Kling, R., & Scacchi, W. (1982). The web of computing: Computing technology as social organization. In M. Yovits (Ed.), *Advances in computers* (Vol. 21, pp. 1-90). New York: Academic.

Malone, T. W. (1985). Designing organizational interfaces. *Proceedings of the CHI'85 Conference on Human Factors in Computer Systems,* 66-71. New York: ACM.

Malone, T. W., Grant, K. R., Turbak, F., Brobst, S. A., & Cohen, M. D. (1987). Intelligent information-sharing systems. *Communications of the ACM, 30,* 390-400.

Nygaard, K. (1986). Program development as a social activity. In H-J. Kugler (Ed.), *Information processing 86* (pp. 189-198). New York: Elsevier North Holland.

Nygaard, K., & Sørgaard, P. (1985). The perspective concept in informatics. *Proceedings of the Aarhus 1985 Working Conference on Development and Use of Systems and Tools* (pp. 75-102). Aarhus University, Aarhus, Denmark.

Orr, J. (1986). Narratives at work, story telling as cooperative diagnostic activity. *Proceedings of the Conference on Computer-Supported Cooperative Work,* 62-72. Austin, TX: MCC.

Schneider, L., & Howard, R. (1985). *Office automation in a manufacturing setting.* Unpublished manuscript prepared for the United States Office of Technology Assessment.

Searle, J. R. (1969). *Speech acts.* Cambridge, England: Cambridge University Press.

Searle, J. R. (1975). A taxonomy of illocutionary acts. In K. Gunderson (Ed.), *Language, mind and knowledge* (pp. 344-369). Minneapolis: University of Minnesota Press.

Stefik, M., Foster, G., Bobrow, D. G., Kahn, K., Lanning, S., & Suchman, L. (1987). Beyond the chalkboard: Computer support for collaboration and problem solving in meetings. *Communications of the ACM, 30,* 32-47.

Suchman, L. (1987). *Plans and situated actions: The problem of human-machine communication.* Cambridge, England: Cambridge University Press.

Winograd, T. (1985). Moving the semantic fulcrum. *Linguistics and Philosophy, 8,* 91-104.

Winograd, T., & Flores, F. (1986). *Understanding computers and cognition: A new foundation for design.* Norwood, NJ: Ablex.

HCI Editorial Record. First manuscript received October 1, 1986. Revision received April 20, 1987. Accepted by Thomas Malone. —*Editor*
