Uber and Language/Action Theory

Michael Arnold Mages
Carnegie Mellon University
michaelam@cmu.edu
DOI: 10.21606/drs.2016.293

Abstract: Mediated communication is the way that distributed and proximate work teams communicate, and is structured nearly completely through software. Users request and make commitments, collaborate on and complete projects, and develop new software systems through software-based conversations. Yet, software designers and developers approach designing conversation software as a series of generic submissions, rather than as an iterative and reflexive process of specific and varied types of speech-acts. This paper examines two pieces of software: The Coordinator and the Uber Partner (driver) app, and a summary of the dialog surrounding the release of the Coordinator as an implementation of Language/Action Theory.

Keywords: conversation; Language/Action theory; systems design; behavior shaping

1. Introduction

Digital communications and workflow management software have become thoroughly embedded in the workplace. Workers communicate through email, text messaging, through software-based services like Basecamp, Slack, custom intranet-based discussion tools, wikis. With the office fragmenting into isolated units of at-home employees, mobile offices, and third-place offices, mediated communications are the way that work gets done. Work conversations take place increasingly through mediated, and media-based experiences. Yet designers and developers lack a critical approach (or even a socially-oriented approach) when designing and prototyping conversation technologies. In my experience developing internet-based software, and helping create online communities, the conversation centers more around the technical scope of what is buildable, the human-resources scope of what is accomplishable, and the user-experience scope of what is acceptable and useable. During a development process, far more effort is spent discussing these factors, and the questions of how people will use the system in a social sense are left for the users to muddle through. Designers of conversational systems choose from the same set of limited and limiting set of
patterns. Yet challenges arise when the design of these conversational systems structure communications and commitment-making in ways that serve interests other than the agents involved in the conversation or commitment. Unfortunately when this aspect is considered, the design of computer systems is all too often directed against the best interests of the user (q.v. darkpatterns.org), and represents an empathy directed away from users and towards value-extraction on behalf of the owners of the software system.

This paper will examine Terry Winograd and Fernando Flores’ approach to designing for conversation and commitment with the approach of Language/Action Theory, and will principally discuss two pieces of software: Winograd and Flores’ Coordinator, and a Language/Action Theory based analysis of the Uber Partner (Driver) mobile phone app.

2. Understanding Computers and Cognition

Language/Action Theory
The debate around the conversational organization of workflow management tools begins in the book *Understanding Computers and Cognition*, with the creation of Language/Action Theory, proposed by Flores and Winograd as new direction for the development of computer software generally, and specifically the problems of creating an Artificial Intelligence.

*Understanding Computers and Cognition* proposes an approach to designing computer systems that abandons the cognitive psychology approach of making computers think like human beings. Winograd and Flores propose approaching the design of computer systems from a perspective that is founded in biology and philosophy rather than what they consider to be a misdirected attempt to make computers replicate human behavior.

The approaches proposed in *Understanding Computers and Cognition* place language at the center of the understanding of computer systems, and propose that computers are more useful as a communication tool, that the principal activity of computer systems design should be to support human activity, rather than to give the machine a kind of agency. In fact, for Winograd and Flores “Nothing exists except through language.” (Winograd & Flores, 1986, p.68)

Winograd and Flores examine commitment and action through *speech acts*. According to J.L. Austin, a speech act is essentially: language (using only literal meaning, not Grice-ian implicature238) in the context of conversation, interpreted as action by people (Austin, 1962). Flores and Winograd’s approach to action-oriented conversation runs a parallel track to Austin: “an understanding of language as meaningful acts by speakers in situations of shared activity.” (Winograd & Flores, 1986, p.54)

---

238 H.P. Grice (1975) details the concept of implicature, essentially the idea that words can imply a meaning that exists outside of their literal meaning in natural language. Statements made with implicature depend upon a broad shared cultural agreement of inferences that contain the meaningful content.
Winograd and Flores pay special attention to a set of commitment-making speech acts. These acts form the foundation for a set of structured commitments that a person could make in a conversation. These acts are:

Flores & Winograd (1986, p58), quoting Searle’s taxonomy of J.L. Austin’s performatives:

- **Assertives** – commit the speaker to something being the case (this I believe)
- **Directives** – attempt to get the hearer to do something (a question is a type of directive, attempts to get the hearer to make an assertion)
- **Commissives** – commit the speaker to do something
- **Expressives** – expresses a psychological state about a situation (apologizing and praising)
- **Declarations** – establish correspondence between the propositional content of speech and reality (pronounce a couple married)

This taxonomy describes what the speaker can do with their utterance, how a person can take action through their language. Flores and Winograd call to attention that these speech acts make sense principally in relation to a conversational background. This conversational background may include the containing culture, a shared history of the participants, an understanding of the current situation. When there is a breakdown in the conversation, it is the inappropriateness or un-relation of the background that is often to blame. (Winograd and Flores say that this is one kind of instance when a listener will think the speaker cannot be taken seriously.)

The following diagram, also from Understanding Computers and Cognition, delineates the abstracted structure for a network of Searle’s performative speech acts, directed towards conditions of satisfaction. Winograd and Flores admit that this kind of reductionist, rational approach is antithetical to the approach they are advocating, but that the goal of their theorizing is to be able to build computer systems that accommodate human action. Computer programming architecture itself is dependent upon the existence of abstracted logical structures. Therefore, some concession must be made in order to have a functional computer program.
Flores and Winograd detail how, through a series of requests, promises, assertions and declarations, a pair of actors can move from irresolution to resolution. Through a conversation for action, one actor can create conditions of satisfaction for the other.

Winograd and Flores clearly state that symbolic language, while important, is not how knowledge exists for human beings. However, representation is key to knowledge sharing. Representation is also key to conversation, as it is through the exchange and manipulation of symbols that conversation can occur, that a being can gain access to another’s understandings, perspectives and approaches. It is through the sharing of symbols that we can communicate, and take action based upon those communications. For Winograd and Flores, language is community property, not personal property.

Yet, it is this challenge of the digitization of the inherently non-digital processes that creates the problem for designing software that can support behaviors that are compatible with, as Lucy Suchman (1994) says, the specificity, heterogeneity and practicality of organizational life as social humans. It is the designer’s obligation to encode sensitivity to the user’s background into a designed solution by developing a rich, historically and culturally informed understanding of the context of the conversation and folding that knowledge into the designed object (Collins, 2004; Collins & Evans, 2007).

**Importance of Conversation in Artificial Intelligence**

Winograd and Flores also describe the problem of attempting to design systems that simulate a conversation with another person where the two share a background. In Alan Turing’s view, the development of seamless human computer interaction in the medium of natural language conversation is a more useful prospect to explore than the question of whether computers can think (Turing, 1950). The compelling simplicity of Turing’s vision of understanding machine ability through conversation led sociologist Harry Collins to propose a taxonomy of expertise to include *interactional expertise*, which can be summarized as the ability to pass as an expert in a certain domain through conversation (Collins, 2004). Collins’
interactional expertise argues against Heidegger’s sense of embodied knowing – that understanding cannot be developed without embodiment. Collins proposes that interactional expertise is achieved when a researcher (like a sociologist or ethnographer) researching a subculture acquires enough of an understanding of the subject domain of their research, they have mastered the language of that expert group. Collins, like Turing, points to the ability to carry on a conversation in a topic area as the key act of fluency of interactional expertise.

Winograd claims that even though language is not fundamental to knowing, because language is our main social tool, language is the way to make a command and the way that commitments are negotiated. Therefore, for Winograd and Flores, the computer cannot be the expert, or behave as an actor in the conversational system, but it can facilitate the communication, can structure and can share the communication. (Winograd & Flores, 1986, p.77) Meaning and Language remain social constructions, but computer will never be an embodied, social being. Therefore our meaning and our language remain inaccessible to it.239

Yet, Jaap Jelsma (2003), following an idea developed by Bruno Latour, tells us that computers, as a component of the socio-technical landscape, can exert a scripting influence on the people who use them. Though it is not embodied, through the shaping and manipulation of symbols, the computer becomes part of the conversational environment, or the site for the conversation, rather than being an actor in the conversation.

Deliberation and Commitment

Though the term “deliberation” is used in many contexts, for Winograd and Flores, moving from irresolution to resolution is deliberation, a kind of conversation. Deliberative conversation is a guided, or facilitated experience that results in action:

1. At some moment in the process of articulating the claims, some incipient partial proposals can be discerned, as different people give opinions, suggestions, disparagements, counter-offers, etc. In this conversation, distinctions between means and goals, parts and wholes are discarded in favor of interpretations about possible causal links, potential results, and inconveniences.

2. At some moment, a sedimented opinion about possible courses of action to be evaluated and considered may begin to appear; this is when the process called ‘choosing’ could be considered. However, the name ‘choosing’ is inadequate, because it suggests algorithmic procedures for selecting the course of action. (Winograd & Flores, 1986, p. 149)

Winograd and Flores note that resolution is “the exploration of a situation, not the application of habitual means” (1986, p. 150). To give an answer for the problem of ‘choosing’ mentioned above, one might turn to anthropologist Annemarie Mol and her investigation of diabetic patients in The Logic of Care (2008). In this work, Mol contrasts a

239 John Searle refers to this as the Chinese Room problem of understanding intelligence: a person, with proper instruction in the presentation of sheets of paper with written Chinese could carry on a written conversation by presenting the sheet as instructed without actually knowing Chinese, and thus pass the Turing Test.
logic of choice with a logic of care, suggesting that there is a kind of discipline, or
mindfulness in care, perhaps the kind of discipline that Winograd and Flores hoped to
gender with The Coordinator. Perhaps Winograd and Flores, had Mol’s articulation of care
been available to them at the time, would have said that The Coordinator exists not to
discipline and structure, but to make employees mindful of the ways in which they are
communicating, and make and receive commitments in the work situation in a more
conscious way.

• The Coordinator

Winograd details the creation of The Coordinator, a piece of software that structures and
tracks commitment among business colleagues (Winograd, 1987). The Coordinator is built
using the theoretical principles of the Language/Action Theory proposed in Understanding
Computers and Cognition. Utilizing Searle’s taxonomy, The Coordinator reifies the different
speech acts as structured forms that correspond to the types of speech acts.

As it is described in Winograd’s paper, initiating a conversation in The Coordinator works as follows:

A user initiates a request, by selecting a request type from a predetermined list of options.
The type of request determines a structured template that will be used to formulate the
request. Below is the example request initiation screen.

| CONVERSE |
|----------------------------------|----------------------------------|
| OPEN CONVERSATION FOR ACTION      | REVIEW / HANDLE                  |
| Request                          | Read new mail                    |
| Offer                            | Missing my response              |
|                                  | Missing other’s response         |
| OPEN CONVERSATION FOR POSSIBILITIES| My promises/offers               |
| Declare an opening                | My requests                      |
|                                  | Commitments due: 24-May-88       |
| ANSWER                           | Conversation records             |
| NOTES                             |                                  |

Figure 2, Converse screen from the Coordinator (Winograd, 1987)

In a system similar to email usage, in the following request screen the user can choose the
recipient, those who will receive copies, as well as a more open tagging and categorization
structure. The request itself includes a subject, a free-form text body as well as three dates
to provide structured timing for the request: a respond-by date, a complete-by date, and an
alert date.
The recipient of the request reviews received requests through the menu shown below:

```
SPEAKING IN A CONVERSATION FOR ACTION

Acknowledge   Promise
Free-Form     Counter-offer
Commit-to-commit Decline
Interim-report Report-completion
```

Figure 3. Request response menu from the Coordinator (Winograd, 1987)

The response screen allows several types of response, core to the original idea are promise, counter-offer, decline. The key element of The Coordinator system is the conversational metadata about the types of requests that are initiated, and the types of responses. While the Coordinator never became a significant part of workflow management, it did engender significant academic dialog around the political aspect of categories, structuring of communication, and the role of digitization of communication processes.

Figure 4. Basecamp HQ workflow management overview screen (basecamphq.com)
Further, the influence of concepts central to The Coordinator can be seen in a number of workflow-tracking applications. As can be seen in the image of 37 Signals’ Basecamp software above, similar features exist for structuring message content, and generating metadata: Message, To-do list, event, and file are more free-form ways to structure a conversation. Similar approaches can be found in other online workflow tracking softwares like Podio (podio.com/site/features/task-management), Siasto (www.siasto.com/tour), Trello (trello.com/tour), Asana (asana.com/product), and others. While I have not traced the genealogy of these applications to The Coordinator, the introduction of workflow software via Flores and Winograd’s offering and the consideration of categories of communication represent a liminal point in the design of workflow information systems.

• Conversations for Action

Independently of Winograd, Fernando Flores wrote Conversations for Action and Collected Essays (2012). In this book, Flores delineates several archetypal conversations and components of these conversations as they occur in business settings. Flores’ underlying assumption in Conversations for Action is that: people are generally trying to act positively; the system of hierarchy in the workplace is fundamentally benevolent; the goal of the conversation is known or at least knowable; there are challenges that get in the way of clear communication; and that people want to ameliorate those challenges.

Flores details 3 types of conversation:

- **for action** - making commitment (previously reified as The Coordinator)
- **for possibilities** - making shared frames
- **for moving forward** - avoiding making characterizations that limit futures

And their underlying challenges:

- **characterizations** - necessary to getting work done, a kind of assessment
- **moods** - when characterizations become assumed and become the underlying context for new assessments.
- **trust** - built up over time: composed of sincerity, competence, reliability, engagement (Flores, 2012)

In Flores’ approach to conversation, characterizations are assessing-type statements that we make about a person or situation, assigning them to a certain typology. For Flores, making characterizations is a dangerous game. Characterizations about people or groups limit future possibilities for working together. However, it is a human act to make characterizations about people and situations. People continuously make characterizations about themselves as well as about others. Flores claims that frequently, people will make characterizations that are not well-grounded. A well-grounded characterization is supported by experience, by a pattern of assertions that one has experienced (Flores, 2012, p. 56). Characterizations are a limiting factor for future action. Carol Dweck describes these characterizations as a type of mindset (Dweck, 2008b), and has done extensive research on the limits for growth and
learning potential that mindset can contribute (Dweck, 2008a). Strangely enough, Flores’ critique of characterization roughly parallels Suchman’s critique of software-structured categorization of utterances.

Interestingly, in the realm of characterization/mindset, there is a relatively new product, Crystal (crystalknows.com), a service that attempts to provide context and shaping for a conversation, and communicate to facilitate a potential email recipient’s receptiveness (also conveniently available as a plugin to Gmail and LinkedIn!). Crystal creates a set of characterizations by scanning publicly available social media profiles. To consider Flores’ and Dweck’s approach to characterization/mindset, the use of this application, while purported to be a boon to communication, easing the interaction with a potential partner, is inherently future-limiting. Besides my concern that, if this software becomes popular, I will be
inundated with emoticon-filled emails (:-P) the provenance of the characterizations offered here is specious, and hidden, and the suggestions given are at the specificity level of horoscope writing at best. Flores states that through characterizations we commit to speculations about the future, and choose to direct where where will discover future opportunities. Basing a conversation from weakly or poorly grounded characterizations limits the development of future opportunities and domains of action.

3. The debate: Do Categories Have Politics?

In an incisive critique of the use and nature of categorization within Flores and Winograd’s Coordinator titled *Do Categories Have Politics?* (1994) anthropologist Lucy Suchman centers on the issue of categorization, and brings forth critiques citing Harvey Sacks, and Michel Foucault. Suchman’s critique engendered such further interest, that a complete issue of Computer-Supported Cooperative Work (1994, vol 3) was devoted to exploring the space of the debate.

Predominantly, Suchman’s critique concerns the nature of categories themselves, and the political nature of who constructs those categories. Borrowing from Sacks’ writing on teen hotrodders, Suchman says “…the adoption of speech act theory as a foundation for system design, with its emphasis on the encoding of speakers' intentions into explicit categories, carries with it an agenda of discipline and control over organization members' actions.” (Suchman, 1994) Suchman points to Michel Foucault’s writing on the training of 18th century soldiers, analogizing the technical workers to Foucault’s soldiers subjected and practiced bodies.

Suchman takes further issue with the schematic nature of model of the conversation for action upon which The Coordinator is based: that the conversation delineated by the model is findable anywhere, a totalizing influence that shuts off the potential for other conversations to exist outside its schema. Suchman’s concern here is that the parametrization of the heterogeneity of work-life is inadequate to serve the art of collaborative work. Suchman argues that The Coordinator serves only to reproduce and reinforce the dominant paradigm of management upon the social order of the workplace. In this article, Suchman presents The Coordinator as a tool for accountability and accountancy, a way for management to track and measure employee productivity, and enforce discipline.

•Categories, Disciplines, and Social Coordination

In a rebuttal, Winograd answers some of Suchman’s critique (Winograd, 1994). Winograd explicitly paints the picture of the proletarian struggle that he claims is the subtext to Suchman’s critique of Language/Action Theory and The Coordinator. Yet, Winograd claims that Suchman has unjustly subjected Language/Action Theory and The Coordinator to oversimplified dichotomies that deny the richness of the social interactions described by the theory.
Winograd agrees with Suchman’s point regarding the inadequacy of The Coordinator to capture the heterogeneity of commitment in workplace life, claiming accurately, that the nature of designing a framework for use in computer systems necessitates a significant degree of abstraction. The development of software architecture privileges recurrent patterns rather than heterogeneity. One of the most significant challenges of software design is to create an algorithmic process that supports a wide range of creative acts. Even something as seemingly simple and pro forma as e-commerce systems (unknown in the days of these papers) inherit radical heterogeneity when they begin to intersect with the systems that resist digitization: multiple overlapping political boundaries that determine sales tax calculation, and systems to facilitate order picking from inventory, packaging, and shipping. Digitization imposes its own discipline upon these acts, as every customer address and tax and shipping profile, every inventory location and every inventory quantity must be coded in a structured system, and e-commerce software must, at the least, communicate with the software system and human beings that manage those inventory locations and quantities, and interface with the bizarre system of multiple, overlapping political and geographic boundaries to calculate sales tax and order shipping based upon the customer’s physical address entries.

While aspects of this argument have become less interesting over the years since these arguments were posed (that our computer-supported collaborative work will intersect with multiple systems, that accountancy of Language/Actions will produce a deluge of data of commitment requests, and will require that hours of work time are spent managing the digital artifacts of those selfsame Language/Action commitments) the core questions: What happens when human processes are digitized? and To what degree is behavior-shaping through technology ethical, and desirable? are still valid questions to be asked as we embed our ethics in our soft- and hardware.

4. Uber

Let’s examine another process, similar to The Coordinator, that may have less benevolence and be more susceptible to Suchman’s critique. An hour of leisurely browsing of http://uberpeople.net/ reveals a plethora of anecdotal evidence of a proletarian struggle against an oppressive bourgeoisie. Some of the discussion threads read as clear as an indictment of Uber’s labor practices as Anna Sewell’s accounts of animal abuse in the Livery industry in Victorian England (Sewell, 1870).

A person working as an Uber driver, is making commitments with customers following the principles of Language/Action theory, yet, the provider of the system (Uber) is explicitly seeking to derive as much profit from the livery as possible, create an experience that is high-quality for the customers and owned by the Uber organization, isolate the livery from remuneration outside the system, while providing only minimal support, and disavowing a committed relationship between the employer and employee.
Following is one critical path through the Uber driver’s app, as dramatized in Uber’s employee training video (archived at https://youtu.be/JvEFw2AGLOw). The training video is 13:35 long, and contains multiple sections, including a basic orientation to the software, details of the customer’s rating module, and instructions on how to use the software and Uber service as a driver. In the following scenario, according to Flores and Winograd’s Conversation for Action model, the customer is A, and the driver, B.
Figure 6: **A: Request (1→2)** The Uber driver receives a request. However, the software hides the scope of the request. The driver is merely notified of the pickup request and location. It does not tell the driver the customer’s desired destination and assumes the customer is actually ready to be picked up.

Figure 7: **B: Promise (2→3)** The Uber driver has the option to not accept the pickup, but unbeknownst to drivers, non-acceptance of too many pickups (drivers are not told how many) will result in the driver’s exile from the Uber network.

Figure 8: **B: Reject, A: Withdraw (2→8)** An Uber driver has the option to cancel a pickup after it has been accepted. The driver may cancel, but must provide metadata about the nature of the cancellation. The metadata determines whether the customer will be charged for the cancellation. Yet, some customers get a number of free cancellations.

Figure 9: **B: Assert (3→4)** Once the driver accepts the customer’s request, the customer is sent a notification via the Uber application. Once calculated, an estimated arrival time is also sent via the Uber application.
Figure 10: **B: Assert (3→4)** Once the driver has arrived at the pickup point, they click the “Arriving Now” button, which generates a text message to the customer. In the training video, potential drivers are cautioned that the customer may not be ready or even at the pickup point, and the driver should wait, but not call the customer, as customers find calls to be annoying.

**A: Declare (4→3) or A: Withdraw (4→9)** The Uber training video does not refer to using the application while in transit, except to suggest that the driver may want to purchase a supplemental standalone GPS hardware, or use Waze or Google Maps. Here, the declare step is invoked when the customer accepts the ride. The customer may also choose to withdraw at this point, paying a $5 penalty for cancellation.

Figure 11: **B: Assert (3→4)** At the end of the trip, the driver presses the “End Trip” button, which ends the trip and generates a fare payment on the customer’s account.

Figure 12: **A: Declare (4→5)** Both the driver and the customer are notified of the full fare amount, and each have the opportunity to rate the experience of the other. However, the driver nets 70–80% of the fare, depending upon the number of riders they have fulfilled in the week.
The Uber application violates several of the best practices that Winograd and Flores set forth for the design of applications that support speech acts. Most salient as relates to the Uber driver’s app, are the repeated violations of the clear delineation of responsibility and scope of promise in commitment. The Uber app and the Uber employee training video do their best to obscure responsibility on behalf of the Uber organization, and diffuse the scope of request on behalf of the customer. To quote Flores and Winograd:

> Once we recognize the machine as an intermediary, it becomes clear that the commitment inherent in the use of language is made by those who produce the system. In the absence of this perspective it becomes all too easy to make the dangerous
mistake of interpreting the machine as making commitments, thereby concealing the source of responsibility for what it does. (Winograd & Flores, 1986, p.155)

By not articulating the scope of the driver’s commitment when the customer’s pickup request is delivered, Uber denies the driver the opportunity to refuse pickups that may be less profitable – such as those pickups that lead the driver to a remote area where acquiring new fares is unlikely. The customer’s rating of drivers is emphasized at throughout the video as a feedback tool for drivers, so they can understand how to create a better customer experience. Yet, Uber maintains and tabulates a set of hidden and poorly defined offences by drivers: declining a pickup request, arriving at a pickup point too slowly, not logging enough time on the system. These offences are tabulated by Uber's system, and a obscure number of them will result in the driver’s exile from the Uber system. Other than customer reviews, the existence of these other ratings of performance are not made available to the driver through any feedback channel, visual or otherwise. The existence of productivity measures aside from the customer’s rating of drivers is made evident in the training video only by oblique statements such as this one at 6:20 “Because all requests go to the closest driver, your acceptance rate is important.” The clearest statement regarding exile from the Uber network is an equivocation made at 1:37 and repeated at 9:30 as regards the customer’s rating of drivers: “If your rating falls below rider expectations, you may lose access to the Uber application.” and this is associated visually with a 1-star rating, perhaps implying that 3 and 4 star ratings are part of the spectrum of acceptable service. Also note, this one-star rating is shown disassociated with the actual Uber application.

That this disciplinary structure exists, but is not visualized or otherwise made evident through the Uber software is an ethical lapse: the aforementioned concealing of responsibility, or more to the point, intentionally obfuscating information that is used to measure employee productivity, and will be used to discipline an employee.

Suchman’s critique of the Coordinor draws out fair points as regards the Uber driver critical path: Language/Action Theory is not sufficient to ensure positive acts in the software.
However, for the exploitation to occur, the keepers of the system of control and discipline must be in a position to effectively execute exploitation. While Winograd and Flores did not attain that scope in the creation of The Coordinator, creators of valuable networks like Uber, Ebay, AirBnB, and Alibaba have the potential to exert oppressive force upon their employees, customers and suppliers through the structuring of commitment, and how their interfaces both conceal and reveal data, and what measures are chosen and shared with employees to track productivity and customer satisfaction. Suchman points to Winograd and Flores’ Coordinator as a tool with an agenda of discipline and control, although Suchman does not go so far to call it a paternalistic application. Paternalism entails responsibility for those whose freedom is restricted. And yet, the Uber application is not even paternalistic. Uber commands without accountability.

5. References


**About the Authors:**

**Michael Arnold Mages** is a PhD candidate at Carnegie Mellon University, School of Design. His doctoral research involves understanding how images, spaces and objects shape high-stakes conversations.